



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



**IMPLEMENTASI *CHATBOT* INFORMASI GAME
BERBASIS *NAMED ENTITY RECOGNITION* DAN
*CONDITIONAL RANDOM FIELD***

SKRIPSI

MUHAMMAD FAJRUL FALAH

2207412019

**POLITEKNIK
NEGERI
JAKARTA**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN
KOMPUTER**

POLITEKNIK NEGERI JAKARTA

2026



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



**IMPLEMENTASI *CHATBOT* INFORMASI GAME
BERBASIS *NAMED ENTITY RECOGNITION* DAN
*CONDITIONAL RANDOM FIELD***

SKRIPSI

**Dibuat untuk Melengkapi Syarat-Syarat yang Diperlukan untuk
Memperoleh Diploma Empat Politeknik**

MUHAMMAD FAJRUL FALAH

2207412019

**POLITEKNIK
NEGERI
JAKARTA**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN
KOMPUTER**

POLITEKNIK NEGERI JAKARTA

2026



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini:

Nama : Muhammad Fajrul Falah
NIM : 2207412019
Jurusan/Program Studi : Teknik Informatika & Komputer / Teknik Informatika - CCIT
Judul Skripsi : IMPLEMENTASI CHATBOT INFORMASI GAME BERBASIS NAMED ENTITY RECOGNITION DAN CONDITIONAL RANDOM FIELD

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Depok, 1 Juli 2026



Falah

Muhammad Fajrul Falah
NIM 2207412019



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

LEMBAR PENGESAHAN

Skripsi diajukan oleh:

Nama : Muhammad Fajrul Falah
NIM : 2207412019
Jurusan/Program Studi : Teknik Informatika & Komputer / Teknik Informatika - CCIT
Judul Skripsi : IMPLEMENTASI CHATBOT INFORMASI GAME BERBASIS NAMED ENTITY RECOGNITION DAN CONDITIONAL RANDOM FIELD

Telah diuji oleh tim penguji dalam Sidang Skripsi pada Hari Rabu, Tanggal 17, Bulan Juni, Tahun 2026, dan dinyatakan **LULUS**.

Disahkan oleh

Pembimbing I : Euis Oktavianti, S.Si., M.T.I.

Penguji I : Dr. Ir. Dewi Yanti Liliana, S.Kom., M.Kom.

Penguji II : Rizki Elisa Nalawati, S.T., M.T.

Penguji III : Chairul Fikri, S.Kom., M.Kom.

Mengetahui :

Jurusan Teknik Informatika dan Komputer

Ketua

Dr. Anita Hidayati, S. Kom., M. Kom.

NIP. 197908032003122003



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas berkat, rahmat, dan karunia-Nya, penulis dapat menyelesaikan laporan skripsi ini dengan baik. Penyusunan laporan skripsi ini dilaksanakan dalam rangka memenuhi salah satu syarat akademik untuk mencapai gelar Diploma Empat (D4) pada program studi di Politeknik Negeri Jakarta. Penulis menyadari bahwa laporan ini tidak akan terwujud tanpa bantuan dan dukungan berbagai pihak. Oleh karena itu, penulis ingin menyampaikan terima kasih kepada:

1. Kedua orang tua penulis yang telah memberikan dukungan moral dan material selama penyusunan skripsi ini;
2. Ika Puspitasari dan Ayuningdyah Dewanti, kakak penulis yang selalu memberikan motivasi serta panutan selama proses penyusunan skripsi;
3. Ibu Dr. Anita Hidayati, S. Kom., M. Kom., selaku Ketua Jurusan Teknik Informatika dan Komputer;
4. Ibu Euis Oktavianti, S.Si., M.T.I., selaku Dosen Pembimbing yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan arahan, saran, serta bimbingan yang sangat berharga dalam penyusunan laporan skripsi ini;
5. Rekan-rekan sejawat dan sahabat yang telah banyak memberikan motivasi, bantuan teknis, dan dukungan selama proses menyelesaikan laporan skripsi ini;

Penulis menyadari bahwa penulisan skripsi ini masih jauh dari kata sempurna. Oleh karena itu, penulis sangat mengharapkan segala bentuk kritik, saran dan masukan yang membangun. Semoga penelitian ini dapat memberikan manfaat bagi pembaca sekalian.

Depok, 1 Juli 2026

Muhammad Fajrul Falah



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya bertanda tangan dibawah ini:

Nama : Muhammad Fajrul Falah
NIM : 2207412019
Jurusan/Program Studi : Teknik Informatika & Komputer / Teknik
Informatika – CCIT

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul:

IMPLEMENTASI *CHATBOT* INFORMASI GAME BERBASIS NAMED ENTITY RECOGNITION DAN CONDITIONAL RANDOM FIELD

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini Politeknik Negeri Jakarta Berhak menyimpan, mengalih mediakan/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Depok, 1 Juli 2026



Muhammad Fajrul Falah
NIM 2207412019



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

IMPLEMENTASI *CHATBOT* INFORMASI GAME BERBASIS *NAMED ENTITY RECOGNITION* DAN *CONDITIONAL RANDOM FIELD*

ABSTRAK

Mekanisme permainan *Genshin Impact* sering kali mengunci akses suatu wilayah hingga pemain menyelesaikan quest prasyarat tertentu. Pencarian panduan akses wilayah secara manual dinilai kurang efisien akibat masukan bahasa pemain yang tidak terstruktur, penggunaan singkatan informal, serta rentan terhadap kesalahan ejaan. Penelitian ini mengimplementasikan *chatbot* penyedia informasi prasyarat quest menggunakan pendekatan *Natural Language Processing (NLP)*. Pemodelan dibangun melalui metode *Named Entity Recognition (NER)* berbasis algoritma *Conditional Random Field (CRF)* untuk mengekstrak entitas lokasi dari teks percakapan. Model ini dipadukan dengan teknik augmentasi data untuk memperkaya variasi bahasa. Sistem diintegrasikan menggunakan metode *Waterfall* dengan layanan back-end *Flask* dan antarmuka web *Laravel*. Hasil pengujian menunjukkan bahwa model *CRF* dengan augmentasi data pada rasio pembagian 80:20 berhasil mencapai tingkat akurasi 99,84% dan *Weighted F1-Score* 99,37%. Pengujian *Black Box* mencatatkan tingkat kelayakan operasional mutlak sebesar 100% pada seluruh skenario uji. Evaluasi *System Usability Scale (SUS)* memperoleh skor rata-rata 83 yang menempatkan sistem pada kategori *Excellent (Grade "A")*. Temuan ini menyimpulkan bahwa *chatbot* yang dikembangkan tangguh, layak digunakan, dan memberikan kemudahan bagi pemain dalam mencari informasi prasyarat quest secara presisi.

Kata Kunci: *Chatbot, Conditional Random Field, Named Entity Recognition, Natural Language Processing*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR ISI

SURAT PERNYATAAN BEBAS PLAGIARISME.....	i
LEMBAR PENGESAHAN	ii
KATA PENGANTAR	iii
ABSTRAK.....	v
DAFTAR ISI.....	vi
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan dan Manfaat.....	4
1.5 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Open-World Game	6
2.2 Chatbot	6
2.3 Software Development Life Cycle	6
2.3.1 Waterfall	7
2.4 Cross Industry Standard Process – Data Mining (CRISP-DM)	11
2.4.1 Business Understanding.....	11
2.4.2 Data Understanding	12
2.4.3 Data Preparation	12
2.4.4 Modeling.....	12
2.4.5 Evaluasi.....	22
2.4.6 Deployment.....	24
2.5 Penelitian Terkait	24
BAB III METODE PENELITIAN.....	26
3.1 Rancangan Penelitian	26
3.2 Tahapan Penelitian	27
3.2.1 Identifikasi Masalah & Studi Literatur	27
3.2.2 Data Preparation	27



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.2.3 Modeling	28
3.2.4 Evaluasi.....	28
3.2.5 Integrasi	28
3.2.6 Testing.....	28
3.2.7 Pelaporan	29
3.3 Sumber Data	29
3.3.1 Populasi.....	29
3.3.2 Sampel	29
3.3.3 Instrumen	29
3.4 Teknik Pengumpulan Data	29
3.4.1 Studi Literatur	29
3.4.2 Observasi	30
3.5 Teknik Analisis Data	30
BAB IV HASIL DAN PEMBAHASAN	32
4.1 Analisis Kebutuhan Sistem	32
4.1.1 Analisis Kebutuhan Fungsional.....	32
4.1.2 Analisis Kebutuhan Non-Fungsional.....	33
4.2 Perancangan Sistem.....	34
4.2.1 Use Case Diagram	34
4.2.2 Activity Diagram	35
4.2.3 Sequence Diagram	38
4.2.4 Class Diagram.....	39
4.2.5 Wireframe	40
4.3 Implementasi Model.....	41
4.3.1 Business Understanding.....	41
4.3.2 Data Understanding	41
4.3.3 Data Preparation	42
4.3.4 Modeling	50
4.3.5 Evaluation	64
4.3.6 Deployment.....	69
4.4 Implementasi Sistem	71
4.4.1 Implementasi Chatbot	71
4.5 Pengujian Sistem	76
4.5.1 Deskripsi Pengujian	76



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.5.2 Prosedur Pengujian	77
4.5.3 Data Hasil Pengujian	80
4.5.4 Analisis Hasil Pengujian.....	84
BAB 5 PENUTUP	86
5.1 Kesimpulan.....	86
5.2 Saran	87
DAFTAR PUSTAKA	88
DAFTAR RIWAYAT HIDUP.....	92
LAMPIRAN.....	93





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 1. Penelitian Terkait	24
Tabel 2. Deskripsi Label pada Dataset.....	42
Tabel 3. Contoh Kalimat pada Dataset	42
Tabel 4. Parameter Pelatihan Model CRF	56
Tabel 5. Hasil Pengujian Model CRF dengan Augmentasi Data.....	64
Tabel 6. Hasil Pengujian Model CRF Tanpa Augmentasi Data	65
Tabel 7. Kerangka System Usability Scale	77
Tabel 8. Kerangka Pengujian Black Box	79
Tabel 9. Hasil Pengujian System Usability Scale	81
Tabel 10. Perhitungan Skor System Usability Scale.....	81
Tabel 11. Grade & Rating SUS Score.....	82
Tabel 12. Data Hasil Pengujian Black Box.....	82

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 2.1 Perwujudan NER.....	13
Gambar 2.2 <i>Linear Chain CRF</i>	14
Gambar 2.3 <i>Confusion Matrix</i>	22
Gambar 3.1 Bagan Rancangan Penelitian.....	26
Gambar 4.1 <i>Use Case Diagram</i> Paimon Guide	34
Gambar 4.2 <i>Activity Diagram</i> Mengakses Menu Chatbot	35
Gambar 4.3 <i>Activity Diagram</i> Melakukan Dialog Dengan Chatbot	36
Gambar 4.4 <i>Activity Diagram</i> Mengelola Profil.....	37
Gambar 4.5 <i>Sequence Diagram</i> Chatbot Paimon Guide.....	38
Gambar 4.6 <i>Class Diagram</i> Database area_prequisites	39
Gambar 4.7 <i>Wireframe</i> Paimon Guide.....	40
Gambar 4.8 <i>Import Library</i> Pendukung.....	43
Gambar 4.9 Inisialisasi Variabel Global dan Mapping Alias	43
Gambar 4.10 Memuat dan Merapihkan Data Terminologi.....	44
Gambar 4.11 Hasil <i>Flattening Gazetteer</i>	44
Gambar 4.12 Proses Tokenisasi Teks, Normalisasi, dan POS Tagging.....	45
Gambar 4.13 Contoh Hasil <i>Tokenization</i> dan <i>Normalization</i>	46
Gambar 4.14 Ekstraksi Fitur dalam <i>Feature Engineering</i>	46
Gambar 4.15 <i>Parsing Dataset</i> Berformat IOB	48
Gambar 4.16 Proses Augmentasi Data (<i>Entity Replacement</i>).....	49
Gambar 4.17 Hasil Augmentasi Dataset	49
Gambar 4.18 Fungsi Perhitungan <i>Node Potential & Edge Potential</i>	50
Gambar 4.19 Fungsi Perhitungan <i>Forward Pass</i>	51
Gambar 4.20 Fungsi Perhitungan <i>Backward Pass</i>	51
Gambar 4.21 Perhitungan Fungsi Normalisasi	52
Gambar 4.22 Perhitungan Probabilitas Marginal <i>Node & Edge</i>	52
Gambar 4.23 Inisialisasi Parameter Untuk Menghitung <i>Gradien</i>	53
Gambar 4.24 Perhitungan Fitur Empiris Untuk <i>Node</i> dan <i>Edge</i>	54
Gambar 4.25 Perhitungan <i>Gradien</i> Berdasarkan Probabilitas <i>Marginal</i>	54
Gambar 4.26 Implementasi <i>Negative Log-Likelihood</i> dan Regularisasi L2	55
Gambar 4.27 Perhitungan <i>Viterbi Decoding</i>	56



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.28 <i>Fuzzy String Matching</i>	57
Gambar 4.29 Inisialisasi Data untuk <i>White-Box Testing</i>	58
Gambar 4.30 Hasil Perhitungan Fitur Aktif Dengan Bobot.....	59
Gambar 4.31 Perhitungan <i>Node Potential</i>	59
Gambar 4.32 Perhitungan <i>Potential Edge</i>	60
Gambar 4.33 Perhitungan <i>Forward Pass</i>	60
Gambar 4.34 Perhitungan <i>Backward Pass</i>	61
Gambar 4.35 Perhitungan Probabilitas <i>Marginal</i>	62
Gambar 4.36 Perhitungan <i>Viterbi Decoding</i>	63
Gambar 4.37 <i>Confusion Matrix</i> Test Size 80:20.....	66
Gambar 4.38 <i>Learning Curve</i> CRF dengan <i>Test Size</i> 80:20	67
Gambar 4.39 Potongan Kode app.py	69
Gambar 4.40 Mengirim <i>User Input</i> 'text' ke Flask.....	70
Gambar 4.41 Ekstraksi <i>Database</i> Untuk Quest Prasyarat	70
Gambar 4.42 Tampilan Awal Paimon Guide	71
Gambar 4.43 Mengirim Sapaan "hi"	72
Gambar 4.44 Profil dan Nickname Berubah	72
Gambar 4.45 User Mencari <i>Region</i>	73
Gambar 4.46 User Mencari Area Spesifik	74
Gambar 4.47 User Menanyakan Dengan Istilah	74
Gambar 4.48 User Menanyakan Nama Area Dengan <i>Typo</i>	75
Gambar 4.49 User memasukkan area diluar <i>Database</i>	76



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1 Latar Belakang

Genshin Impact merupakan *open-world action role-playing video game* yang dikembangkan oleh HoYoverse. Permainan ini berlatar di dunia fantasi luas bernama Teyvat, yang terdiri dari berbagai wilayah dengan karakteristik ekosistem dan elemen yang unik (Genshin Impact Wiki, 2026b). Sebagai salah satu game dengan pertumbuhan konten yang masif, Genshin Impact berfokus pada pengalaman eksplorasi dan penyelesaian cerita melalui berbagai jenis misi, mulai dari misi utama hingga misi dunia (*World Quest*) yang sangat beragam dan tersebar di seluruh peta permainan (Genshin Impact Wiki, 2025).

Salah satu aspek fundamental dalam progresivitas permainan ini adalah sistem eksplorasi (*Exploration*), di mana pemain didorong untuk berinteraksi dengan lingkungan dan membuka wilayah baru guna mendapatkan sumber daya serta mengungkap cerita dunia (Genshin Impact Wiki, 2026a). Karakteristik *Open World* dalam Genshin Impact menerapkan mekanisme aksesibilitas yang kompleks. Sejumlah wilayah atau zona tertentu sering kali terkunci dan baru dapat diakses setelah pemain menyelesaikan *World Quest* tertentu, yang mana penyelesaian misi ini sering mengubah lanskap area permainan secara permanen (Genshin Impact Wiki, 2025).

Proses pencarian informasi prasyarat secara manual ini dinilai kurang efisien untuk menjaga kelancaran eksplorasi pemain, diperlukan inovasi chatbot yang interaktif untuk menyajikan data lokasi dan urutan misi secara instan. *Chatbot* yang berbasis aturan terbatas dalam hal fleksibilitas interaksi. Sementara di sisi lain, sistem berbasis kecerdasan buatan (*Artificial Intelligence*) menggunakan pemrosesan *natural language* untuk memahami percakapan dan pesan pengguna yang rumit secara mendalam (Illescas-Manzano *et al.*, 2021).

Chatbot perlu dapat menggunakan model *tagging* seperti *Hidden Markov Models* (HMM) untuk memproses data dengan benar dalam menemukan elemen dalam



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

teks. Model ini mencoba menebak label yang ada berdasarkan kondisi suatu entitas pada saat ini, akan tetapi ditemukan masalah karena HMM memperlakukan bagian-bagian ini secara terpisah, yang menyulitkan untuk memahami arti kalimat yang kompleks. *Conditional Random Fields* (CRF) mengatasi hal ini dengan memungkinkan dalam memahami arti sebuah kalimat tanpa dibatasi oleh bagaimana fitur-fitur tersebut saling berhubungan. CRF telah terbukti membantu memperbaiki masalah struktur kalimat dan menemukan kata-kata baru yang sebelumnya tidak diketahui oleh sistem (Potočár dan Kvet, 2023).

Kurangnya kejelasan prasyarat pembukaan area di dalam game membuat pemain harus mencari panduan secara manual melalui berbagai komunitas daring. Tantangan yang muncul kemudian adalah kompleksitas bahasa pengguna saat bertanya, mereka cenderung menyebutkan nama lokasi secara bervariasi di dalam kalimat bahasa alami yang kompleks dan tidak terstruktur. Tidak terstrukturnya bahasa ini seperti dalam penggunaan singkatan tidak standar, variasi penyebutan nama lokasi, hingga pencampuran istilah informal yang mencampur batas antar entitas dalam kalimat. Hambatan signifikan dalam pemrosesan teks tersebut seringkali bersumber dari keberadaan kata yang tidak dikenali (*unknown words*) yang mencakup nama diri (*proper nouns*), akronim, serta dinamika penambahan kosakata baru dalam struktur bahasa (Potočár dan Kvet, 2023).

Diperlukan pendekatan *Natural Language Processing* (NLP), khususnya metode *Named Entity Recognition* (NER). Penerapan NLP memungkinkan sistem untuk memahami permintaan pengguna dalam bahasa alami secara lebih akurat dan memberikan respons yang relevan (Simorangkir *et al.*, 2024). Metode NER menawarkan solusi efektif dalam mengekstraksi entitas informasi krusial, seperti lokasi atau kategori spesifik, dari data teks tidak terstruktur tanpa ketergantungan pada format input yang baku. Pemanfaatan NER pada sumber informasi digital terbukti signifikan dalam mengidentifikasi entitas vital yang mendukung proses pengambilan keputusan secara cepat dan akurat (Karo Karo, Dewi dan Syahrin, 2025).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Penelitian ini membutuhkan metode NER berbasis algoritma CRF ke sistem chatbot sehingga dapat secara otomatis mendapatkan entitas utama seperti nama area dari percakapan pemain. Implementasi ini diharapkan dapat mengatasi masalah bahasa yang tidak terstruktur dan kesulitan dalam mencari urutan misi dengan menggunakan mekanisme Exploration Progress. Hal ini akan memastikan bahwa informasi prasyarat akurat dan respons penyajian data cepat dalam membantu kelancaran eksplorasi pemain.

1.2 Perumusan Masalah

Bagaimana mengimplementasikan chatbot informasi quest prasyarat Genshin Impact berbasis *Named Entity Recognition (NER)* menggunakan algoritma *Conditional Random Field (CRF)* dalam mengekstrak entitas lokasi dan nama quest serta menyajikan informasi prasyarat secara akurat?

1.3 Batasan Masalah

Agar penelitian ini lebih terarah dan fokus pada tujuan utama, ditetapkan batasan masalah sebagai berikut:

1. Sistem yang dibangun merupakan aplikasi berbasis situs web (*web-based*) yang menggunakan *Framework* Laravel sebagai antarmuka pengguna dan manajemen data, serta bahasa pemrograman Python sebagai layanan *back-end* untuk pemrosesan kecerdasan buatan.
2. Studi kasus dibatasi secara spesifik pada pencarian informasi *quest* prasyarat (*prerequisite quest*) di dalam game Genshin Impact.
3. Lingkup informasi yang diberikan oleh sistem dibatasi secara eksklusif pada daftar *quest* atau kondisi yang harus diselesaikan untuk membuka area tertentu. Sistem tidak menyediakan panduan penyelesaian *quest* (*walkthrough*), maupun tips bermain.
4. Pemrosesan bahasa alami pada *chatbot* dibatasi hanya untuk menerima dan merespons masukan teks dalam Bahasa Inggris. Pemrosesan ini berfokus secara spesifik pada ekstraksi entitas lokasi di dalam game menggunakan algoritma *Conditional Random Field (CRF)*. *Chatbot* dirancang untuk



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

merespons masukan yang berkaitan dengan pencarian informasi prasyarat area tersebut dan tidak dioptimalkan untuk percakapan bebas (*chit-chat*).

5. *Chatbot* yang dikembangkan hanya berfungsi sebagai agen layanan otomatis berbasis teks (*text-based*) untuk memberikan informasi prasyarat, tanpa fitur pemrosesan suara (*voice recognition*) atau integrasi secara langsung dengan akun pemain di dalam game (*in-game integration*).

1.4 Tujuan dan Manfaat

Tujuan utama dari penelitian ini adalah mengimplementasikan aplikasi chatbot penyedia informasi *quest* prasyarat Genshin Impact berbasis *Named Entity Recognition* (NER) menggunakan algoritma *Conditional Random Field* (CRF) yang terintegrasi dengan sistem informasi berbasis situs web.

Adapun manfaat yang diharapkan dari penelitian ini adalah:

1. Membantu pemain Genshin Impact secara otomatis dan cepat dalam mengetahui urutan *quest* prasyarat (*prerequisite quest*) yang diperlukan untuk mengakses suatu *quest* lanjutan atau area tertentu, sehingga mengurangi kerumitan pencarian informasi secara manual melalui mesin pencari atau wiki.
2. Membuktikan efektivitas penerapan algoritma *Conditional Random Field* (CRF) dalam mengekstrak entitas lokasi (*Named Entity Recognition*) secara akurat dari teks bahasa alami yang diinputkan oleh pengguna.
3. Menyediakan layanan respons otomatis yang beroperasi selama 24 jam penuh, memungkinkan pemain untuk mendapatkan informasi prasyarat kapan saja saat sedang bermain tanpa harus bergantung pada balasan dari forum komunitas atau grup diskusi.
4. Memberikan informasi alur penyelesaian *quest* yang lebih terstruktur dan jelas kepada pemain, sehingga dapat menghemat waktu eksplorasi dan meningkatkan kelancaran pengalaman bermain (*player experience*) secara keseluruhan.

1.5 Sistematika Penulisan

Sistematika Penulisan Skripsi terdiri dari 5 bab, yaitu:



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1. BAB I PENDAHULUAN

Bab I terdiri dari latar belakang, perumusan masalah, batasan masalah, tujuan & kegunaan dan sistematika penulisan terkait pembuatan *chatbot*.

2. BAB II TINJAUAN PUSTAKA

Bab II menjelaskan tentang teori-teori yang berkaitan atau digunakan dalam proses pengerjaan implementasi *chatbot* beserta dengan penelitian sejenis yang dapat dijadikan perbandingan.

3. BAB III METODE PENELITIAN

Bab III menjelaskan tahapan dalam perancangan *chatbot* terkait *design diagram* dan metode yang akan digunakan.

4. BAB IV HASIL DAN PEMBAHASAN

Membahas hasil implementasi sistem, evaluasi performa fitur, serta analisis efektivitas dalam mencapai tujuan penelitian.

5. BAB V PENUTUP

Berisi kesimpulan dari penelitian serta saran untuk pengembangan lebih lanjut.

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Open-World Game

Open-world game merupakan genre yang memungkinkan pemain untuk menjelajah secara mandiri, memilih berbagai misi, dan berinteraksi di dalam dunia virtual yang luas. Tingkat kebebasan yang tinggi, ditambah dengan pengalaman bermain yang imersif, telah menjadikan genre *game* ini sebagai sektor penting dalam industri hiburan digital (Ma, 2025).

Untuk mendukung keleluasaan tersebut, *Open-world game* menggunakan struktur tugas yang berlapis, mencakup misi utama (*primary missions*), misi sampingan (*secondary missions*), dan tugas harian (*daily assignments*). Sistem misi yang terperinci ini dirancang khusus untuk mempertahankan keterlibatan pemain dengan memanfaatkan *near-win effect* (efek merasa hampir menang), sekaligus mendorong strategi bermain yang memiliki tujuan jelas (Ma, 2025).

2.2 Chatbot

Chatbot merupakan program berbasis AI (*Artificial Intelligence*) yang dapat berbicara dengan orang-orang dengan cara yang terdengar seperti orang sungguhan, baik melalui teks maupun suara. Teknologi ini digunakan secara strategis sebagai *agent customer service* yang dapat dengan cepat menjawab pertanyaan dan keluhan, yang dapat mempercepat dan meningkatkan layanan (Solikhah dan Ngetich, 2025).

2.3 Software Development Life Cycle

Software Development Life Cycle (SDLC) adalah kerangka kerja terstruktur dalam rekayasa perangkat lunak yang mencakup banyak langkah, seperti menganalisis dan mendefinisikan persyaratan, perencanaan, pengembangan, *testing*, dan pemeliharaan sistem. Metode ini merupakan cara sistematis untuk memastikan bahwa proses pengembangan aplikasi berjalan konsisten dari tahap pengumpulan persyaratan hingga penyerahan kepada pengguna (Hasanah dan Indriawan, 2021).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.3.1 Waterfall

Penelitian ini menggunakan model Waterfall sebagai kerangka dasar dalam membangun sistem. Model ini memiliki siklus hidup perangkat lunak yang linier dan berurutan yang mencakup langkah-langkah analisis, desain, implementasi kode, dan pengujian sistem (Handrianto dan Sanjaya, 2020).

Penerapan metode ini secara bertahap dianggap efektif dalam mengurangi kemungkinan kesalahan operasional, mengingat atribut alur kerja yang terstruktur secara sistematis dengan fokus penyelesaian yang jelas (Hafizhah, Theresiawati dan Warsuta, 2022).

A. Analisis Kebutuhan

Fase analisis kebutuhan merupakan tahapan saat menuliskan spesifikasi persyaratan *software*. Termasuk dalam mencari tahu bagaimana fungsi sistem dan proses bisnis akan bekerja bersama. Dengan secara menyeluruh mengidentifikasi keterbatasan pengembangan, mengevaluasi keandalan dan kerentanan sistem, dan memilih teknologi yang paling sesuai untuk implementasi (Handrianto dan Sanjaya, 2020).

1. Kebutuhan Non Fungsional

a. Visual Studio Code

Visual Studio Code adalah editor kode sumber (*Source Code Editor*) yang kompatibel dengan beberapa sistem operasi, dilengkapi dengan fitur-fitur termasuk debugging, integrasi Git, dan alat refactoring. Perangkat lunak ini sangat serbaguna, memungkinkan pengguna untuk menyesuaikan pengaturannya dan menginstal *extension* tambahan untuk meningkatkan kemampuannya (Azizi *et al.*, 2024).

b. Tailwind

Tailwind CSS adalah *framework* berbasis *utility-first* yang bertujuan untuk mempercepat desain antarmuka pengguna. *Framework* ini memfasilitasi pembuatan desain kustom yang cepat dan mudah dengan menawarkan berbagai *utility class* siap pakai, sehingga menghilangkan kebutuhan untuk pengkodean *Cascading Style Sheets* (CSS) manual (Azizi *et al.*, 2024).



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

c. SQLite

SQLite merupakan *Database Management System* (DBMS) yang dibangun menggunakan bahasa pemrograman C dan beroperasi sebagai mesin basis data SQL yang tertanam langsung di dalam aplikasi. Berbeda dengan mesin basis data berbasis *client-server* seperti MySQL, pengoperasian SQLite tidak membutuhkan sistem *server* terpisah maupun proses instalasi khusus (*zero-configuration*). Sistem ini membaca dan menulis data secara langsung ke dalam bentuk *file* tunggal pada penyimpanan *disk* (Suraya dan Sholeh, 2022).

d. Figma

Figma merupakan *tools* desain berbasis web (*web-based*) yang memudahkan orang untuk berkolaborasi dalam mendesain antarmuka pengguna (*User Interface*) dan pembuatan *prototype* sistem (Fadila *et al.*, 2024).

e. Postman API

Postman adalah platform kolaboratif dalam mengembangkan dan menguji *Application Programming Interfaces* (API). *Software* ini membantu *developer* memeriksa apakah API berfungsi, menangani permintaan HTTP, dan memastikan bahwa data dapat dipertukarkan antar sistem dengan cara yang menjaga semuanya berjalan lancar (Azizi *et al.*, 2024).

2. Kebutuhan Fungsional

Kebutuhan fungsional merupakan spesifikasi teknis yang menjelaskan fasilitas dan aktivitas operasional yang harus dapat dilakukan sistem secara lengkap. Kelompok persyaratan ini mencakup langkah-langkah yang perlu dilakukan *software*, serta informasi yang perlu tersedia dan keluaran data yang perlu dihasilkan sistem untuk memenuhi tujuan pengembangan aplikasi (Setiyani dan Tjandra, 2021).

B. Desain

Tahap desain sistem merupakan fase yang berfungsi untuk mentransformasikan spesifikasi kebutuhan perangkat lunak hasil analisis menjadi model representasi

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

visual atau cetak biru arsitektur sistem. Tahapan ini bertujuan untuk menyusun landasan teknis yang terstruktur mengenai alur proses dan struktur data agar dapat diimplementasikan secara presisi ke dalam bentuk kode program (Handrianto dan Sanjaya, 2020).

1. Activity Diagram

Activity diagram merupakan gambar yang menunjukkan bagaimana suatu sistem bekerja secara sistematis. Diagram ini menunjukkan urutan kejadian dalam sistem, mulai dari memulai suatu proses, mengambil keputusan tentang cara menjalankannya, hingga menyelesaikannya (Narulita, Nugroho dan Abdillah, 2024).

2. Use Case Diagram

Use case diagram adalah model visual yang menunjukkan cara kerja suatu sistem dengan memperlihatkan bagaimana komponen aktor, kasus penggunaan (*Use Case*), dan hubungan antar elemen-elemen tersebut berinteraksi. Hal ini membuat *requirement* perilaku sistem menjadi lebih jelas (Narulita, Nugroho dan Abdillah, 2024).

3. Sequence Diagram

Sequence diagram menunjukkan bagaimana objek dalam sebuah kasus penggunaan (*Use Case*) berinteraksi satu sama lain secara waktu nyata. Diagram ini menunjukkan urutan pengiriman dan penerimaan pesan antar objek dalam dimensi waktu tertentu (Narulita, Nugroho dan Abdillah, 2024).

4. Class Diagram

Class diagram menunjukkan struktur statis suatu sistem, diagram ini juga menunjukkan koneksi logis antar objek, memberikan gambaran lengkap tentang *data architecture system* (Hafizhah, Theresiawati dan Warsuta, 2022).



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

5. User Interface

User Interface (UI) adalah bagian dari sistem yang menghubungkan pengguna dengan *software*. Proses *UI Design* adalah tentang membuat visualisasi yang tidak hanya terlihat bagus tetapi juga mengutamakan kegunaan sehingga *user* dapat dengan mudah mengakses semua fitur sistem (Fadila *et al.*, 2024)

C. Implementasi

Tahap implementasi adalah saat draf desain sistem diubah menjadi kode yang dapat dijalankan. Ini adalah tahap di mana cetak biru arsitektur sistem diubah menjadi perangkat lunak yang berfungsi menggunakan bahasa pemrograman seperti PHP (Handrianto dan Sanjaya, 2020).

1. Website

Website merupakan sekumpulan halaman web yang semuanya berada dalam domain yang sama dan memiliki berbagai jenis informasi multimedia yang dapat diakses secara online. (Arimbi, Kartinah dan Della, 2022).

2. Laravel

Laravel adalah *framework* pengembangan web berbasis PHP yang populer karena memiliki struktur file yang baik dan mudah digunakan. Framework ini memiliki arsitektur *Model-View-Controller* (MVC) yang memudahkan pembuatan aplikasi yang terstruktur dan aman. Selain itu, *Laravel* juga memiliki fitur penting seperti *routing*, modularitas, dan mekanisme migrasi *database* bawaan (Witriyono dan Aipina, 2022).

D. Testing

Tahap pengujian merupakan evaluasi kinerja program untuk menemukan masalah atau bug. Proses ini bertujuan untuk memastikan bahwa sistem berfungsi sebagaimana mestinya sebelum diimplementasi sepenuhnya (Handrianto dan Sanjaya, 2020).

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1. Black Box Testing

Black Box Testing cara untuk memeriksa apakah perangkat lunak berfungsi hanya dengan melihat cara kerjanya dari luar, tanpa melihat kode yang membuatnya berfungsi. Metode ini berfokus pada pengecekan kesesuaian antara data input dan output untuk memastikan sistem bekerja sesuai rencana (Hafizhah, Theresiawati dan Warsuta, 2022).

2. System Usability Scale (SUS)

System Usability Scale (SUS) alat yang dapat diandalkan untuk mengevaluasi kegunaan berbagai *software product*. Alat ini melakukannya dengan menggunakan metode yang cepat dan sederhana. Metode ini menggunakan kuesioner dengan sepuluh pertanyaan dan skala Likert lima poin yang menggabungkan pernyataan positif dan negatif untuk memberikan skor numerik tentang seberapa puas dan menerima pengguna terhadap sistem tersebut (Kosim, Aji dan Darwis, 2022).

2.4 Cross Industry Standard Process – Data Mining (CRISP-DM)

Cross Industry Standard Process for Data Mining (CRISP-DM) merupakan kerangka kerja standar yang banyak digunakan untuk proyek *data mining*. Kerangka kerja ini memiliki struktur siklus hidup enam fase yang memastikan hasil analisis valid dan bermanfaat, mulai dari *Business Understanding* hingga *Model Deployment* (Kurniawan dan Yasir, 2022).

2.4.1 Business Understanding

Tahap *Business Understanding* merupakan langkah pertama dalam menentukan tujuan proyek dari sudut pandang kebutuhan bisnis. Informasi ini kemudian digunakan untuk membuat definisi masalah *data mining* teknis yang dapat digunakan untuk menyusun rencana yang baik untuk mencapai tujuan tersebut (Kurniawan dan Yasir, 2022).



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.4.2 Data Understanding

Fase *Data Understanding* dimulai dengan *data acquisition* dan kemudian berlanjut ke aktivitas eksplorasi untuk mengetahui seberapa baik data tersebut dan mendapatkan wawasan awal yang dapat digunakan untuk membuat hipotesis yang relevan untuk pengujian lebih lanjut (Kurniawan dan Yasir, 2022).

2.4.3 Data Preparation

Data Preparation adalah proses mengubah data mentah menjadi kumpulan data akhir. Ini termasuk memilih atribut mana yang akan dipertahankan, membersihkan data, dan mengubahnya agar sesuai dengan algoritma pemodelan yang digunakan (Kurniawan dan Yasir, 2022).

2.4.4 Modeling

Tahap *Modeling* melibatkan penggunaan beragam teknik *data mining* dan kalibrasi parameter model untuk mencapai hasil optimal, yang seringkali memerlukan penyesuaian selama fase persiapan data agar sesuai dengan spesifikasi format input model (Kurniawan dan Yasir, 2022)

A. Python

Python adalah bahasa pemrograman tingkat tinggi yang banyak digunakan di dunia *software development* karena sangat fleksibel. Bahasa ini memiliki banyak *library* yang dapat digunakan untuk mengimplementasikan algoritma *machine learning* (ML) dan analisis data (Suraya dan Sholeh, 2022).

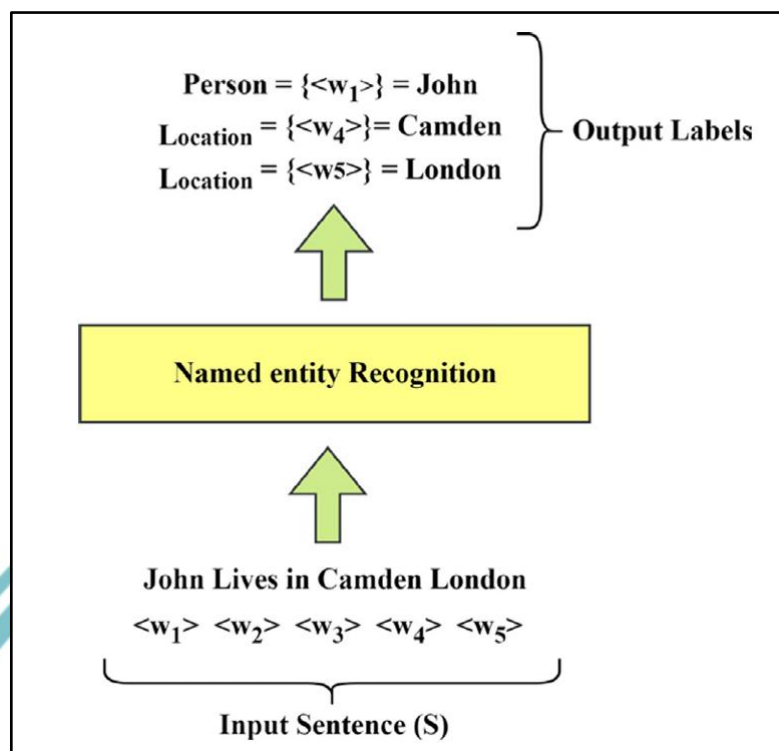
B. Named Entity Recognition

Named Entity Recognition (NER) didefinisikan sebagai mekanisme komputasi yang menemukan dan mengurutkan entitas bernama dalam dokumen teks ke dalam kelas semantik standar, seperti orang, organisasi, atau tempat. Tujuan utama dari proses ini adalah untuk menemukan referensi entitas secara akurat sehingga segmen informasi dapat diubah dari teks mentah menjadi struktur data yang terorganisir (Jehangir, Radhakrishnan dan Agarwal, 2023).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 2.1 Perwujudan NER

Sumber: Jehangir, Radhakrishnan dan Agarwal, 2023

NER adalah langkah penting dalam mendapatkan informasi dari data tidak terstruktur, yang seringkali sulit dilakukan dengan komputer pintar. Manusia dapat melakukan NER dengan cukup mudah karena adanya isyarat visual seperti huruf kapital dalam kata. Namun, tugas ini sangat sulit bagi mesin, yang membutuhkan metode algoritma khusus untuk mengklasifikasikan entitas dari domain terbuka secara otomatis dan akurat (Budi dan Suryono, 2023).

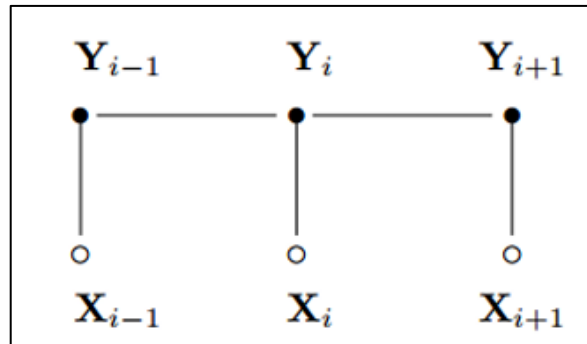
C. Conditional Random Field

Conditional Random Field (CRF) merupakan sebuah model probabilistik yang diakui karena efektivitasnya dalam menangani tugas segmentasi dan pelabelan dalam NLP. Metode ini menggunakan kerangka kerja kondisional yang secara struktural menggabungkan atribut model grafis tak berarah, di mana simpul-simpul grafik menandakan urutan label kelas yang berasal dari data yang diproses (Khan *et al.*, 2022).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 2.2 Linear Chain CRF

Sumber: Jaariyah dan Rainarli, 2017

CRF telah terbukti bekerja dengan baik sebagai metode pelabelan sekuensial untuk menangani kompleksitas struktur linguistik dokumen Indonesia. Metode ini menggunakan berbagai fitur semantik untuk mendapatkan makna kalimat secara utuh, yang secara langsung membantu membuat identifikasi dan klasifikasi entitas bernama menjadi lebih akurat (Santoso *et al.*, 2020).

Algoritma CRF berjalan sebagai sebuah model grafis yang tak berarah, di mana model ini dilatih guna memaksimalkan nilai probabilitas kondisional. Probabilitas kondisional ini menghitung seberapa besar peluang munculnya output label kelas (y) ketika diberikan input observasi (x) (Jaariyah dan Rainarli, 2017). Persamaan probabilitas kondisional tersebut dirumuskan sebagai berikut:

$$p(y|x) = \frac{1}{Z(x)} \prod_c \psi_c(y_c, x) \quad (2.1)$$

Keterangan:

$p(y|x)$: Probabilitas kondisional dari label y terhadap data masukan x .

x : Himpunan data observasi yang berupa kata.

y : Himpunan label kelas entitas yang mungkin dihasilkan.

$Z(x)$: Fungsi normalisasi dari distribusi probabilitas kondisional.

c : Elemen pembentuk fungsi potensial (mencakup *node potensial* dan *edge potensial*).

$\prod_c$: Perkalian seluruh nilai fungsi potensial pada elemen c

$\psi_c(y_c, x)$: Fungsi potensial yang menghubungkan label data dengan fitur.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Agar probabilitas dari semua urutan state menjadi bernilai satu, digunakan konstanta normalisasi yang disimbolkan dengan $Z(x)$. Konstanta normalisasi ini dihitung melalui fungsi berikut:

$$Z(x) = \sum_y \prod_c \psi_c(y_c, x) \quad (2.2)$$

Keterangan:

$Z(x)$: Fungsi normalisasi dari distribusi probabilitas kondisional.

$\sum_y$: Penjumlahan dari seluruh kemungkinan kombinasi himpunan label y .

$\prod_c$: Perkalian seluruh nilai fungsi potensial pada elemen c .

$\psi_c(y_c, x)$: Fungsi potensial yang menghubungkan label data dengan fitur.

y_c : Label kelas entitas pada elemen c .

x : Data masukan yang berupa kata.

c : Elemen pembentuk fungsi potensial (mencakup *node potensial* dan *edge potensial*).

CRF mendefinisikan hubungan antar elemen yang disebut sebagai fungsi potensial. Fungsi potensial ini terdiri dari dua komponen utama, yaitu *Node Potential* dan *Edge Potential*.

Node Potential (ϕ) merupakan perhitungan probabilitas yang berfungsi untuk merepresentasikan nilai asosiasi antara fitur kelas (label y) dengan data masukan (observasi x). Persamaan untuk *Node Potential* adalah sebagai berikut:

$$\phi_t(y_t, x) = \exp(\sum_k \lambda_k f_k(y_t, x, t)) \quad (2.3)$$

Keterangan:

$\phi_t(y_t, x)$: Nilai *node potential* ke- t

$\exp$: Fungsi eksponensial.

$\sum_k$: Penjumlahan seluruh fitur *edge* ke- k yang aktif.

λ_k : Nilai parameter atau bobot untuk fitur *node* ke- k .

$f_k(y_t, x, t)$: Fitur *node* ke- k pada posisi ke- t .



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

y_t : Label kelas pada kata yang sedang diproses.

x : Data masukan yang berupa kata.

t : Indeks posisi kata pada data masukan.

Nilai $f_k(y_t, x, t)$ di dalam persamaan *node potential* (2.3) bertindak sebagai fungsi fitur *state* yang akan menghasilkan nilai *boolean* (1 atau 0). Fitur ini akan bernilai 1 apabila label kelas y_t memiliki asosiasi yang cocok dengan karakteristik kata x pada posisi tersebut, dan bernilai 0 jika sebaliknya. Contohnya:

$$f_k(y_t, x, t) = \begin{cases} 1 & \text{if } y_t = \text{B-LOC}, x = \text{is_location_connector} \\ 0 & \text{otherwise} \end{cases}$$

Edge Potential (ψ) digunakan untuk menilai asosiasi antara label kelas untuk kata yang saat ini sedang diproses dengan label kelas untuk kata berikutnya. Hubungan transisi ini dirumuskan dengan:

$$\psi_t(y_t, y_{t+1}, x) = \exp(\sum_{k'} \lambda_{k'} f_{k'}(y_t, y_{t+1}, x, t)) \quad (2.4)$$

Keterangan:

$\psi_t(y_t, y_{t+1}, x)$: Nilai *edge potential* ke- t .

$\exp$: Fungsi eksponensial.

$\sum_{k'}$: Penjumlahan seluruh fitur *edge* ke- k' yang aktif.

$\lambda_{k'}$: Nilai parameter atau bobot untuk fitur *edge* ke- k' .

$f_{k'}(y_t, y_{t+1}, x, t)$: Fungsi fitur *edge* ke- k' pada posisi ke- t .

y_t : Label kelas pada kata yang sedang diproses.

y_{t+1} : Label kelas pada kata berikutnya.

x : Data masukan yang berupa kata.

t : Indeks posisi kata pada data masukan.

Nilai $f_{k'}(y_t, y_{t+1}, x, t)$ di dalam persamaan *edge potential* (2.4) bertindak sebagai fungsi fitur *transition* yang juga menghasilkan nilai boolean (1 atau 0). Fitur ini akan bernilai 1 apabila transisi atau perpindahan dari label kelas kata saat ini y_t menuju label kelas kata berikutnya y_{t+1} dianggap valid, dan bernilai 0 jika tidak valid. Berikut Contohnya:



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

$$f_{k'}(y_t, y_{t+1}, x, t) = \begin{cases} 1 & \text{if } y_t = O, y_{t+1} = B - \text{LOC} \\ 0 & \text{otherwise} \end{cases}$$

Untuk mendapatkan nilai yang paling optimal pada parameter fitur *node* dan fitur *edge*, CRF menggunakan tahap *training* dengan memaksimalkan nilai probabilitas kemiripan pada data latih menggunakan *maximum likelihood*. Persamaan *log-likelihood* yang digunakan adalah:

$$\mathcal{L} = \sum_{t \in [1, T]} \sum_k \lambda_k f_k(y_t, x, t) - \sum_{t \in [1, T]} \sum_{k'} \lambda_{k'} f_{k'}(y_t, y_{t+1}, x, t) - \log Z(x) - \sum_k \frac{\lambda_k^2}{\sigma^2} \quad (2.5)$$

Keterangan:

$\mathcal{L}$: Nilai dari maksimum *likelihood* (*log-likelihood*).

T : Jumlah kata keseluruhan di dalam satu kalimat masukan.

λ_k : Nilai parameter atau bobot untuk fitur *node* ke- k .

$f_k(y_t, x, t)$: Fitur *node* ke- k pada posisi ke- t .

$\lambda_{k'}$: Nilai parameter atau bobot untuk fitur *edge* ke- k' .

$f_{k'}(y_t, y_{t+1}, x, t)$: Fungsi fitur *edge* ke- k' pada posisi ke- t .

$Z(x)$: Fungsi normalisasi probabilitas untuk kalimat masukan x .

σ : Standar deviasi yang berfungsi sebagai penalti regularisasi.

$\sum_{t \in [1, T]}$: Total akumulasi nilai fungsi sepanjang rentang urutan $T = 1$ sampai T .

Turunan pertama dari *log-likelihood* (nilai gradien) digunakan untuk memperbarui parameter bobot pada tiap iterasi. Gradien untuk fitur *node* disimbolkan dengan G_k^x sedangkan untuk fitur *edge* disimbolkan dengan $G_{k'}^x$

$$G_k^x = \sum_{t \in [1, T]} (f_k(y_t, x, t) - \sum_{y_t} P_t(y_t | x) f_k(y_t, x, t)) - \frac{\lambda_k}{\sigma^2} \quad (2.6)$$

Keterangan:

G_k^x : Nilai gradien ke- k untuk fungsi fitur *node*.

T : Jumlah kata keseluruhan di dalam satu kalimat masukan.

$f_k(y_t, x, t)$: Fitur *node* ke- k pada posisi ke- t .

$P_t(y_t | x)$: Probabilitas *node* ke- t (peluang label y_t terhadap masukan x).



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

y_t : Label kelas pada kata yang sedang diproses.

x : Data masukan yang berupa kata.

λ_k : Nilai parameter atau bobot untuk fitur *node* ke- k .

σ : Standar deviasi yang berfungsi sebagai penalti regularisasi.

$$G_{k'}^x = \sum_{t \in [1, T]} (f_{k'}(y_t, y_{t+1}, x, t) - \sum_{y_t} \sum_{y_{t+1}} P_t(y_t, y_{t+1} | x) f_{k'}(y_t, y_{t+1}, x, t)) - \frac{\lambda_{k'}}{\sigma^2} \quad (2.7)$$

Keterangan:

$G_{k'}^x$: Nilai gradien ke- k' untuk fungsi fitur *edge*.

T : Jumlah kata keseluruhan di dalam satu kalimat masukan.

$f_{k'}(y_t, y_{t+1}, x, t)$: Fitur *edge* ke- k' pada posisi ke- t .

$P_t(y_t, y_{t+1} | x)$: Probabilitas *edge* ke- t (peluang transisi dari *label* y_t ke y_{t+1} terhadap masukan x).

y_t : Label kelas pada kata yang sedang diproses.

y_{t+1} : Label kelas pada kata berikutnya.

x : Data masukan yang berupa kata.

$\lambda_{k'}$: Nilai parameter atau bobot untuk fitur *edge* ke- k' .

σ : Standar deviasi distribusi Gauss

Untuk menghitung peluang probabilitas di dalam perhitungan gradien, sistem memanfaatkan metode pemrograman dinamis bernama *forward-backward pass* yang bertugas menelusuri keseluruhan urutan data.

Forward Pass (α), proses ini menelusuri data dari awal (indeks $t = 1$) hingga akhir (indeks T). Persamaannya untuk $t = 1$ adalah:

$$\alpha_1[y_1] = 1/S \quad (2.8)$$

Keterangan:

$\alpha_1[y_1]$: Nilai *forward pass* pada urutan pertama $t = 1$.

S : Jumlah label.

Sedangkan untuk $t \geq 2$ persamaannya adalah:

$$\alpha_t[y_t] = k_t \sum_{y_{t-1}} \alpha_{t-1}[y_{t-1}] \phi_t(y_t, x) \psi_{t-1}(y_{t-1}, y_t, x) \quad (2.9)$$



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Keterangan:

$\alpha_t[y_t]$: Nilai *forward pass* urutan ke- t .

k_t : *Scaling factor* untuk menormalisasi nilai probabilitas ke- t sehingga

$$\sum_{y_t} \alpha_t[y_t] = 1, k_t > 0$$

$\sum_{y_{t-1}}$: Penjumlahan seluruh kemungkinan label pada urutan sebelumnya.

$\alpha_{t-1}[y_{t-1}]$: Nilai *forward pass* pada urutan ke- $t - 1$

$\phi_t(y_t, x)$: Nilai *node potential* ke- t

$\psi_{t-1}(y_{t-1}, y_t, x)$: Nilai *edge potential* ke- t (Transisi dari urutan $t - 1$ ke t).

Backward Pass (β), proses ini melakukan penelusuran mundur dari akhir (T) menuju awal (1).

Persamaannya untuk $t = T$ adalah:

$$\beta_T[y_T] = 1/S \quad (2.10)$$

Keterangan:

$\beta_T[y_T]$: Nilai *backward pass* pada urutan terakhir (T).

S : Jumlah label.

Sedangkan untuk $t < T$ persamaannya adalah:

$$\beta_t[y_t] = \mu_t \sum_{y_{t+1}} \beta_{t+1}[y_{t+1}] \phi_{t+1}(y_{t+1}, x) \psi_t(y_t, y_{t+1}, x) \quad (2.11)$$

Keterangan:

$\beta_t[y_t]$: Nilai *backward pass* urutan ke- t .

μ_t : *Scaling factor* untuk menormalisasi nilai probabilitas ke- t sehingga

$$\sum_{y_t} \beta_t[y_t] = 1, \mu_t > 0$$

$\sum_{y_{t-1}}$: Penjumlahan seluruh kemungkinan label pada urutan sebelumnya.

$\beta_{t+1}[y_{t+1}]$: Nilai *backward pass* pada urutan ke- $t + 1$.

$\phi_{t+1}(y_{t+1}, x)$: Nilai *node potential* pada urutan ke- $t + 1$.

$\psi_t(y_t, y_{t+1}, x)$: Nilai *edge potential* ke- t (Transisi dari urutan t ke $t + 1$).

Kalkulasi probabilitas lokal yaitu probabilitas *node* dan probabilitas *edge* dilakukan setelah proses penelusuran *forward pass* dan *backward pass* selesai dikerjakan.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Untuk menentukan probabilitas *node* yakni peluang kemunculan label kelas kata y_t berdasarkan data masukan x , digunakan perumusan berikut:

$$P_t(y_t|x) = \omega_t \alpha_t[y_t] \phi_t(y_t, x) \beta_t[y_t] \quad (2.12)$$

Keterangan:

$P_t(y_t|x)$: Nilai probabilitas *node* pada urutan ke- t .

ω_t : Faktor normalisasi untuk memastikan $\sum_{y_t} P_t(y_t|x) = 1$.

$\alpha_t[y_t]$: Nilai *forward pass* urutan ke- t .

$\phi_t(y_t, x)$: Nilai *node potential* ke- t

$\beta_t[y_t]$: Nilai *backward pass* urutan ke- t .

Probabilitas *edge* dihitung untuk mengukur peluang kemunculan label y_t pada data x yang secara sekuensial diikuti oleh label y_{t+1} pada kata berikutnya. Persamaan yang digunakan untuk perhitungan probabilitas *edge* adalah:

$$P_t(y_t, y_{t+1}|x) = \gamma_t \alpha_t[y_t] \phi_t(y_t, x) \psi_t(y_t, y_{t+1}, x) \phi_{t+1}(y_{t+1}, x) \beta_{t+1}[y_{t+1}] \quad (2.13)$$

Keterangan:

$P_t(y_t, y_{t+1}|x)$: Nilai probabilitas *edge* pada urutan ke- t .

γ_t : Faktor normalisasi untuk memastikan $\sum_{y_t} P_t(y_t, y_{t+1}|x) = 1$.

$\alpha_t[y_t]$: Nilai *forward pass* urutan ke- t .

$\phi_t(y_t, x)$: Nilai *node potential* ke- t

$\psi_t(y_t, y_{t+1}, x)$: Nilai *edge potential* ke- t

$\phi_{t+1}(y_{t+1}, x)$: Nilai *node potential* pada urutan selanjutnya.

$\beta_{t+1}[y_{t+1}]$: Nilai *backward pass* urutan selanjutnya.

Setelah model selesai dilatih dan nilai-nilai parameter fitur optimal berhasil didapatkan, tahap akhir adalah implementasi (inferensi) terhadap sekumpulan data uji. Penentuan urutan label paling optimal secara keseluruhan dilakukan dengan metode *Viterbi Decoding* yang tersusun atas dua proses utama:

Maximal Forward Pass, Langkah ini dilakukan untuk menemukan probabilitas paling optimal bagi data masukan dengan mempertimbangkan hubungan label kelas kata yang sedang diproses dengan label kelas kata berikutnya.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Persamaannya untuk $t = 1$ adalah:

$$\alpha_1^{max}[y_1] = 1/S \quad (2.14)$$

Keterangan:

$\alpha_1^{max}[y_1]$: Nilai *Viterbi forward pass* (probabilitas maksimum) pada urutan pertama $t = 1$.

S : Jumlah label.

Sedangkan untuk $t \geq 2$ hingga T , persamaanya adalah:

$$\alpha_t^{max}[y_t] = k_t \max_{y_{t-1}} (\alpha_{t-1}^{max}[y_{t-1}] \phi_t(y_t, x) \psi_t(y_{t-1}, y_t, x)) \quad (2.15)$$

Keterangan:

$\alpha_t[y_t]$: Nilai *Viterbi forward pass* (probabilitas maksimum) pada urutan ke- t .

k_t : *Scaling factor* untuk menormalisasi nilai probabilitas ke- t .

$\max$: Operasi untuk mencari nilai maksimum dari seluruh kemungkinan label y_{t-1} pada urutan sebelumnya (y_{t-1}).

$\alpha_{t-1}^{max}[y_{t-1}]$: Nilai *Viterbi forward pass* maksimum yang diperoleh dari urutan ke- $t - 1$.

$\phi_t(y_t, x)$: Nilai *node potential* ke- t

$\psi_t(y_{t-1}, y_t, x)$: Nilai *edge potential* ke- t (Transisi dari urutan $t - 1$ ke t).

Backtracking Pass, model menelusuri jejak kembali ke belakang guna menarik kesimpulan tentang urutan label akhir yang terbaik (paling optimal).

Persamaannya adalah:

$$y_t^* = \arg \max_{y_t} (\alpha_t^{max}[y_t] \phi_t(y_t, x)) \quad (2.16)$$

Keterangan:

y_t^* : Label kelas optimal (terpilih) pada urutan ke- t .

$\arg \max_{y_t}$: Fungsi yang digunakan untuk memilih label y_t yang menghasilkan

nilai tertinggi dari ekspresi di dalamnya.

$\alpha_t^{max}[y_t]$: Nilai *Viterbi forward pass* (probabilitas kumulatif maksimum) urutan ke- $t - 1$ untuk label y .

$\phi_t(y_t, x)$: Nilai *node potential* ke- t untuk label y_t , dan data masukan x .



2.4.5 Evaluasi

Tahap *Evaluation* adalah tentang melakukan pengecekan menyeluruh terhadap model untuk memastikan bahwa model tersebut secara teknis benar dan hasilnya sesuai dengan tujuan bisnis sebelum beralih ke fase implementasi (Kurniawan dan Yasir, 2022).

A. Confusion Matrix

Confusion Matrix merupakan instrumen validasi untuk mengukur kinerja model klasifikasi melalui komparasi antara label prediksi sistem dan label aktual atau *ground truth* (Darwansah, Arifiyanti dan Hadiwiyaniti, 2025). Metode ini memetakan distribusi hasil prediksi ke dalam representasi tabular yang mencakup empat indikator klasifikasi utama yakni *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), serta *False Negative* (FN).

	Prediksi : <i>Positive</i>		Prediksi: <i>Negative</i>	
Aktual : <i>Positive</i>	<i>True Positive</i> (TP)		<i>False Positive</i> (FP)	
Aktual : <i>Negative</i>	<i>False Negative</i> (FN)		<i>True Negative</i> (TN)	

Gambar 2.3 Confusion Matrix

Sumber: Karo Karo, Dewi dan Syahrin, 2025

B. Accuracy

Akurasi adalah rasio yang merepresentasikan seberapa sering model melakukan prediksi dengan benar secara keseluruhan dibandingkan dengan total data yang diuji (Darwansah, Arifiyanti dan Hadiwiyaniti, 2025).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.17)$$

TP = *True Positive*

TN = *True Negative*

FP = *False Positive*

FN = *False Negative*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

C. Precision

Precision merupakan metrik evaluasi yang mengukur tingkat keakuratan model dalam memprediksi kelas positif dengan menghitung rasio antara prediksi positif yang benar terhadap total seluruh prediksi positif yang dihasilkan oleh model (Darwansah, Arifiyanti dan Hadiwiyan, 2025).

$$Precision = \frac{TP}{TP + FP} \quad (2.18)$$

$TP = \text{True Positive}$

$FP = \text{False Positive}$

D. Recall

Recall adalah metrik yang digunakan untuk mengukur kemampuan model dalam menemukan kembali seluruh data yang benar-benar termasuk dalam kelas positif, sehingga meminimalkan risiko adanya data positif yang tidak terdeteksi (Darwansah, Arifiyanti dan Hadiwiyan, 2025).

$$Recall = \frac{TP}{TP + FN} \quad (2.19)$$

$TP = \text{True Positive}$

$FN = \text{False Negative}$

E. F1 Score

F1 Score didefinisikan sebagai tingkat akurasi yang perhitungannya didasarkan pada nilai presisi (*precision*) dan recall dari suatu pengujian (Muhamad Malik Matin, 2023).

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (2.20)$$



2.4.6 Deployment

Tahap *Deployment* adalah tahap akhir, di mana kesimpulan dari proses pemodelan tidak menandakan berakhirnya proyek, melainkan bertindak sebagai kesempatan untuk mengatur dan menyajikan pengetahuan untuk aplikasi pengguna. Proses implementasi pada tahap ini terutama bergantung pada persyaratan, yang bervariasi dari pembuatan laporan dasar hingga prosedur *data mining* yang rumit dan berulang (Kurniawan dan Yasir, 2022).

A. Flask

Flask adalah *framework* Python yang ditujukan untuk mengembangkan aplikasi daring, yang memberikan pengembang kemampuan untuk menggabungkan berbagai *extension* dan modul sesuai kebutuhan. (Suraya dan Sholeh, 2022).

2.5 Penelitian Terkait

Tabel 1. Penelitian Terkait

Judul	Ekstraksi Informasi Bencana Banjir Dari Berita Online Berbasis Named Entity Recognition (Karo Karo, Dewi dan Syahrin, 2025)	Pemodelan Chatbot Rekomendasi Hotel Dengan Menggunakan Natural Language Processing (Simorangkir <i>et al.</i> , 2024)	Hybrid Conditional Random Fields and K-Means for Named Entity Recognition on Indonesian News Documents (Santoso <i>et al.</i> , 2020)
Metode	<i>Named Entity Recognition</i> (NER) berbasis <i>Conditional Random Field</i> (CRF) dengan pelabelan <i>chunking</i> dan <i>IOB Tagging</i>	<i>Natural Language Processing</i> (NLP) menggunakan modul StanfordNLP untuk <i>Intent Recognition</i> dan <i>Named Entity Recognition</i> (NER), serta algoritma kesamaan kosinus (<i>cosine similarity</i>)	Pendekatan Hibrida <i>Conditional Random Fields</i> (CRF) dan algoritma klusterisasi <i>K-Means</i> dengan fitur <i>Word Embedding</i> (Word2Vec)
Hasil	Penggunaan <i>IOB tagging</i> pada CRF meningkatkan kinerja model dengan <i>F1-score</i> sebesar 93.6% untuk	Chatbot berbasis NLP terbukti lebih akurat dalam memahami permintaan pengguna dan lebih efisien dalam memberikan	Model hibrida (CRF + K-Means) memberikan kinerja terbaik pada skenario pembagian data 60-40 dengan <i>F1-</i>

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

entitas kota dan waktu. Metode ini efektif mendukung pengambilan keputusan cepat dalam penanggulangan bencana.	respons dibandingkan metode pencarian manual, serta meningkatkan kepuasan pengguna.	Score 87.18%, lebih unggul sekitar 4.3% hingga 5.01% dibandingkan model <i>baseline</i> dan Bi-LSTM
---	---	---

Dari ketiga penelitian di Tabel 1., kesamaan sistem terdapat pada penelitian Simorangkir *et al.* (2024) yang mengembangkan Chatbot sebagai media interaksi otomatis. Relevansi metode ditunjukkan oleh Karo Karo, Dewi dan Syahrin (2025) yang menerapkan *Named Entity Recognition* (NER) dengan algoritma *Conditional Random Field* (CRF) untuk ekstraksi entitas informasi. Pemilihan metode CRF ini semakin diperkuat oleh penelitian Santoso *et al.* (2020) yang membuktikan efektivitas algoritma tersebut dalam menangani karakteristik linguistik dokumen berbahasa Indonesia secara presisi.

POLITEKNIK
NEGERI
JAKARTA

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

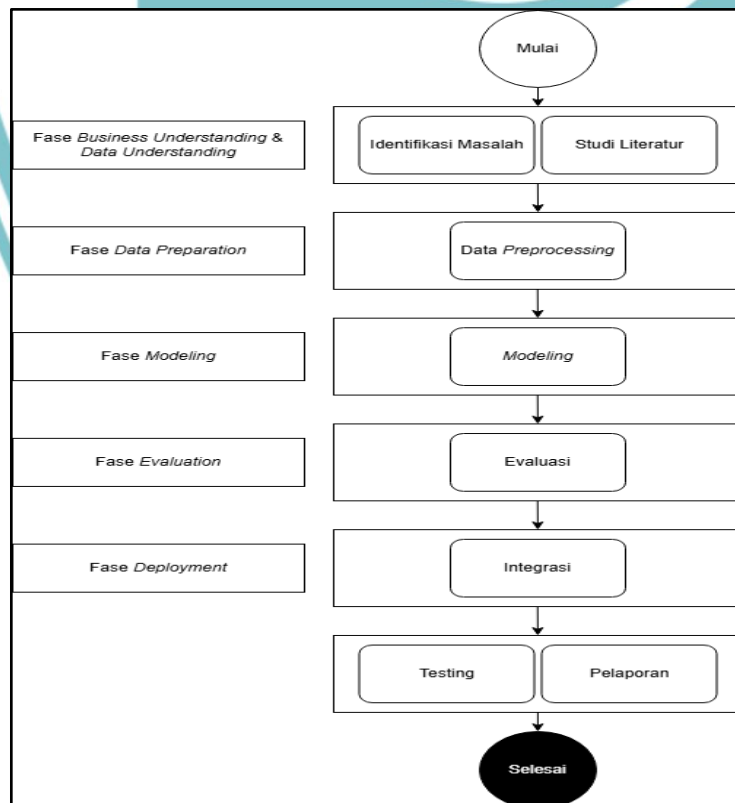
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB III

METODE PENELITIAN

3.1 Rancangan Penelitian

Penelitian yang akan dilakukan adalah pengimplementasian *chatbot* informasi *quest* prasyarat *game* Genshin Impact berbasis *Named Entity Recognition* (NER) menggunakan algoritma *Conditional Random Field* (CRF) untuk mengatasi pencarian informasi manual yang tidak efisien akibat bahasa pengguna yang tidak terstruktur. Model kecerdasan buatan yang dibangun akan menggunakan algoritma CRF untuk mengekstraksi entitas lokasi dari percakapan, yang kemudian diintegrasikan ke dalam aplikasi berbasis *web* menggunakan *back-end* Python untuk logika NER dan *framework* Laravel untuk manajemen antarmuka pengguna. Penelitian ini menggabungkan model pengembangan perangkat lunak *Waterfall* dengan kerangka kerja *data mining* CRISP-DM untuk memastikan validitas model AI dan keandalan sistem dalam memberikan informasi prasyarat secara otomatis. Rancangan penelitian dapat dilihat pada Gambar 3.1.



Gambar 3.1 Bagan Rancangan Penelitian



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.2 Tahapan Penelitian

Tahapan penelitian dalam melakukan implementasi *chatbot* informasi *quest* prasyarat ini mengikuti alur sistematis bagan pada gambar 3.1. Adapun rincian dari setiap prosesnya adalah sebagai berikut:

3.2.1 Identifikasi Masalah & Studi Literatur

Tahap ini mengawali dengan mengidentifikasi kendala pencarian informasi oleh pemain, terutama dalam hal kompleksitas alur *quest* prasyarat pada *game Open World* seperti Genshin Impact yang memiliki mekanisme pembukaan akses untuk lokasi tertentu. Masalah utama yang ditemukan adalah waktu yang kurang efisien akibat pencarian informasi lokasi dan prasyaratnya yang masih dilakukan secara manual. Studi literatur dilakukan untuk memperdalam pemahaman mengenai algoritma CRF, metode NER, serta penelitian terdahulu yang relevan mengenai *chatbot* dan ekstraksi informasi.

3.2.2 Data Preparation

Di dalam proses ini, data percakapan pelanggan yang bersifat tidak terstruktur akan diolah agar siap digunakan dalam pemodelan. Proses ini mengacu pada fase *Data Preparation* dalam CRISP-DM. Langkah-langkah yang dilakukan meliputi:

1. Data Labeling

Memberikan label pada entitas penting seperti Nama Area (*Location*) secara manual menggunakan format IOB (*Inside, Outside, Beginning*).

2. Augmentasi Data

Melakukan proses augmentasi untuk memperkaya variasi data latih, mengingat pelanggan sering menggunakan singkatan tidak standar dan istilah informal. Hal ini dilakukan dengan teknik *Entity Replacement* menggunakan data dari *Wiki Genshin Impact* untuk memastikan model dapat mengenali berbagai nama lokasi dan misi.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.2.3 Modeling

Proses *Modeling* berfokus pada penerapan algoritma *Conditional Random Field* untuk tugas *Named Entity Recognition*. Dipilih CRF karena dapat sepenuhnya memahami konteks kalimat tanpa dibatasi oleh gagasan bahwa fitur-fitur saling independen. Hal ini membuatnya cocok untuk menangani struktur kalimat yang rumit dari pelanggan. Model ini dibangun menggunakan bahasa pemrograman Python dan library yang membantu dalam pembelajaran mesin.

3.2.4 Evaluasi

Efektivitas metode CRF dalam ekstraksi entitas dinilai melalui evaluasi model. Teknik *Confusion Matrix* digunakan untuk memastikan nilai *True Positive*, *False Positive*, *True Negative*, dan *False Negative*, sesuai dengan kriteria evaluasi *Data Mining*. Untuk menjamin prediksi model yang tepat terhadap entitas lokasi meskipun kelas entitas dalam dataset tidak seimbang, pengukuran kinerja mencakup *Akurasi*, *Presisi*, *Recall*, dan *F1-Score*.

3.2.5 Integrasi

Tahap ini merupakan proses penggabungan model AI yang telah dilatih ke dalam arsitektur sistem. Model CRF akan di-*deploy* menggunakan *framework* Flask sebagai layanan API (*Application Programming Interface*). API ini kemudian akan digunakan oleh aplikasi utama yang dibangun menggunakan *framework* Laravel. Laravel akan bertugas mengirimkan *input* teks pelanggan ke Flask, menerima hasil ekstraksi entitas *location*, dan menggunakan entitas tersebut untuk mencari serta menampilkan informasi prasyarat *quest* yang sesuai berdasarkan data yang tersimpan di basis data SQLite.

3.2.6 Testing

Digunakan *Black box testing* mengevaluasi sistem untuk memverifikasi bahwa fitur-fitur beroperasi dengan benar tanpa mengakses arsitektur kode yang mendasarinya. *System Usability Scale* (SUS) digunakan untuk melakukan penilaian



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kegunaan dan mengevaluasi kepuasan serta kemudahan penggunaan bagi pengguna akhir (*end-user*).

3.2.7 Pelaporan

Tahap terakhir pelaporan mencakup pendokumentasian seluruh proses penelitian, mulai dari analisis kebutuhan, desain, implementasi kode, hingga hasil pengujian.

3.3 Sumber Data

3.3.1 Populasi

Populasi data untuk penelitian ini mencakup riwayat obrolan dari pemain pada *server* Discord resmi Genshin Impact (*Genshin Impact Official Discord*) dalam rentang waktu tahun 2021 hingga 2025, beserta terminologi dari permainan Genshin Impact seperti nama lokasi dan nama *quest* yang harus diselesaikan.

3.3.2 Sampel

Dataset yang digunakan sebagai sampel telah dikurasi (*filtered*) menggunakan teknik *purposive sampling*, yang secara spesifik difokuskan pada teks percakapan yang berisi pertanyaan mengenai cara mengakses suatu area atau lokasi pada game Genshin Impact, beserta istilah yang relevan dengan *prerequisite quest*.

3.3.3 Instrumen

Instrumen penelitian mencakup perangkat keras (Laptop) dan *software* (*Visual Studio Code*, Python, SQLite, Postman) yang digunakan untuk mengumpulkan, melabeli, dan memproses data.

3.4 Teknik Pengumpulan Data

Penelitian ini menggunakan dua pendekatan utama dalam pengumpulan data, yaitu studi literatur dan observasi:

3.4.1 Studi Literatur

Dilakukan studi literatur untuk mengumpulkan landasan teori dan referensi yang relevan dengan topik penelitian, dengan penelusuran terhadap jurnal ilmiah dan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

artikel yang membahas tentang *Named Entity Recognition*, algoritma *Conditional Random Field*, serta pengembangan *chatbot*. Informasi yang diperoleh dari studi literatur digunakan untuk memahami konsep dasar, cara kerja algoritma, dan metode evaluasi yang akan diterapkan dalam sistem.

3.4.2 Observasi

Teknik observasi diterapkan pada dua sumber data utama, obrolan dari *server* Discord resmi Genshin Impact dan dokumentasi *game* dari Genshin Impact Wiki. Observasi pada Discord ditujukan untuk membantu dalam memahami bagaimana pemain secara alami menggunakan bahasa, bagaimana mereka menggunakan singkatan non-standar, dan berbagai cara mereka menanyakan cara mengakses suatu lokasi. Analisis data *game* dilakukan untuk menandai bagian-bagian penting, seperti nama lokasi dan kompilasi *prerequisite quest* yang dibutuhkan, yang merupakan syarat utama untuk membuka akses area tersebut. Informasi yang diperoleh dari observasi ini merupakan sumber yang baik untuk memberi label dan menambah data yang dibutuhkan untuk melatih model.

3.5 Teknik Analisis Data

Teknik pengumpulan data, yaitu studi literatur dan observasi, akan menghasilkan suatu wawasan yang bersifat komprehensif yang akan digunakan untuk mengimplementasikan *chatbot* informasi *quest* prasyarat Genshin Impact berbasis *Named Entity Recognition* dan algoritma *Conditional Random Field* pada penelitian ini. Melalui teknik studi literatur akan dilakukan analisis yang difokuskan kepada sintesis informasi yang didapatkan dari berbagai sumber teoritis, hal ini dilakukan untuk dapat memahami konsep *preprocessing* data teks seperti *tokenization* dan ekstraksi fitur linguistik, serta teori pemodelan statistik yang relevan, sehingga setelahnya dapat dilakukan identifikasi masalah terkait ambiguitas istilah yang mungkin muncul dalam percakapan pemain.

Metode observasi akan melibatkan data obrolan dari *server* Discord dan data *game* Genshin Impact, yang mana data tersebut akan dianalisis untuk menentukan pola entitas yang menjadi target ekstraksi, yaitu lokasi. Pada tahap ini, analisis dilakukan



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dengan menggunakan algoritma *Conditional Random Field* untuk menghitung probabilitas label entitas berdasarkan konteks kata di sekitarnya, serta melakukan evaluasi kinerja model menggunakan analisis kuantitatif *Confusion Matrix* (*Precision*, *Recall*, dan *F1-Score*). Teknik analisis terhadap data observasi ini diperlukan untuk memastikan model memiliki tingkat akurasi yang cukup sebelum diintegrasikan.

Hasil dari kedua teknik di atas akan digabungkan dalam analisis sistem. Pengetahuan yang didapatkan dari proses pemodelan statistik dan pemahaman alur pencarian informasi *quest* akan menjadi landasan untuk mengembangkan perangkat lunak menggunakan metode *Waterfall*. Tahapan ini mencakup analisis kebutuhan fungsional dan non-fungsional, yang kemudian dirancang menggunakan diagram UML (*Use Case*, *Activity*, *Sequence*, *Class Diagram*) sebelum akhirnya diimplementasikan ke dalam kode program sistem chatbot yang utuh.

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB IV HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan Sistem

Tahap Analisis Kebutuhan sistem dilakukan untuk mengidentifikasi dan mengumpulkan informasi agar persyaratan sistem chatbot dapat dipahami dengan baik. Analisis kebutuhan pada *chatbot* penyedia informasi *quest* prasyarat ini didasarkan pada hasil studi literatur dan observasi terhadap data obrolan pemain. Setelah mengidentifikasi alur pencarian informasi tersebut, dilakukan rincian analisis kebutuhan sistem yang terdiri dari kebutuhan fungsional dan kebutuhan non-fungsional. Hasil dari analisis kebutuhan sistem ini digunakan sebagai acuan untuk melaksanakan tahapan pemodelan yang meliputi *business understanding*, *data understanding*, *data preparation*, *modeling*, dan *evaluation*, agar model yang dikembangkan sesuai dengan tujuan penelitian.

4.1.1 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional mendefinisikan fitur dan kemampuan utama yang dapat diakses oleh pengguna agar sistem *Chatbot* dapat memberikan informasi prasyarat quest secara efektif. Berikut adalah aspek fungsional yang perlu dipenuhi oleh sistem:

- Pengguna dapat mengirimkan pesan teks berupa pertanyaan mengenai lokasi atau wilayah tertentu di dalam game melalui kolom input chat.
- Pengguna dapat melihat balasan dari chatbot yang berisi informasi detail prasyarat quest.
- Pengguna dapat melihat daftar area yang tersedia di bawah suatu Region jika pengguna memasukkan nama Region tersebut.
- Pengguna menerima pesan kesalahan atau panduan dari chatbot jika input yang dimasukkan tidak mengandung nama lokasi yang dikenali oleh sistem.
- Pengguna dapat melihat riwayat percakapan selama sesi berlangsung di dalam panel chat utama.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.1.2 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional menetapkan standar performa dan spesifikasi teknis agar sistem *chatbot* dapat beroperasi secara stabil. Fokus utamanya adalah memastikan kesesuaian infrastruktur perangkat keras dan perangkat lunak untuk mendukung jalannya model serta aplikasi web secara optimal.

A. Kebutuhan Perangkat Keras

- Processor: Minimal Intel Core i5 Generasi ke-10 atau setara.
- Memori (RAM): Minimal 8 GB.
- Penyimpanan: Minimal memiliki ruang kosong sebesar 1 GB.

B. Kebutuhan Perangkat Lunak

- Sistem Operasi: Windows 10/11 atau Linux.
- Bahasa Pemrograman: PHP 8.3+ dan Python 3.10+.
- Framework: Laravel 13 dan Flask 3.0.
- Library Utama: NLTK, Scipy, dan Vite.
- Database: SQLite.
- Tools: Visual Studio Code, Docker Desktop, dan Git.

C. Kebutuhan Perangkat Lunak (UPA-TIK)

- Environment: Docker.
- Requirement Deployment: Docker Container dan Docker Image.
- Database: SQLite.
- Bahasa Pemrograman: PHP.
- Framework: Laravel.



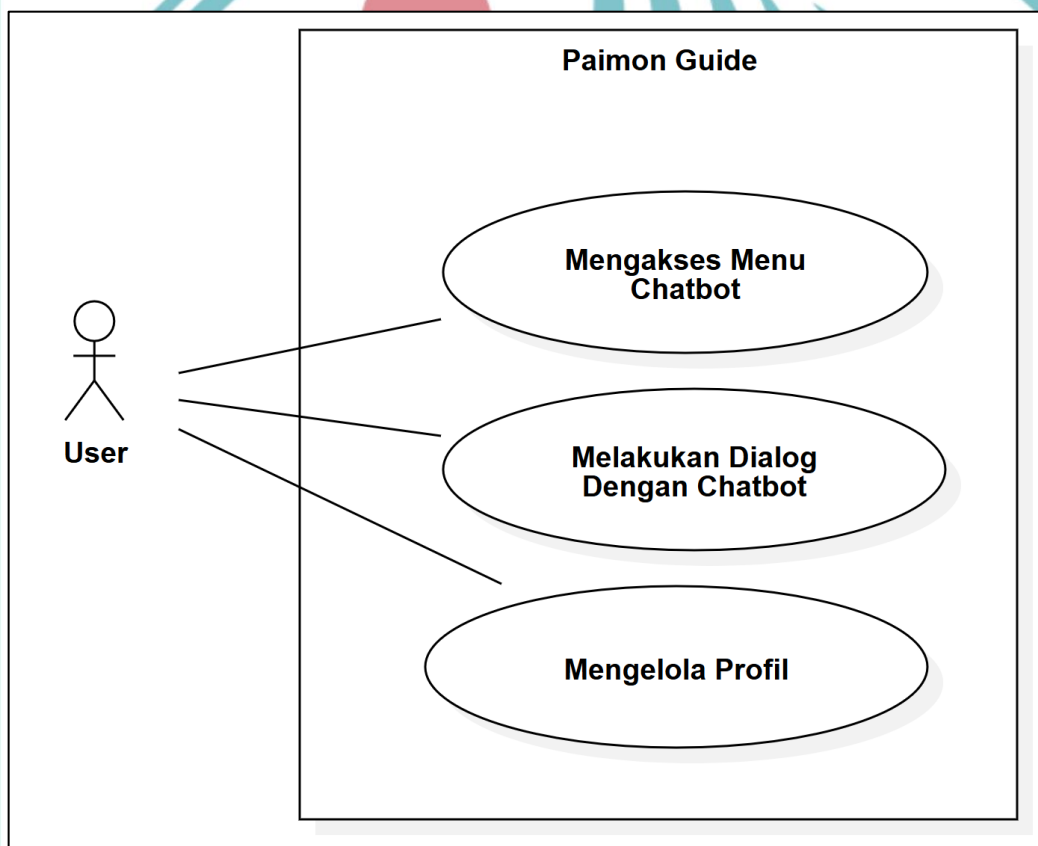
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.2 Perancangan Sistem

Perancangan sistem dilakukan untuk merancang *chatbot* yang akan dibangun berdasarkan kebutuhan yang telah dianalisis sebelumnya. Dalam pengembangan sistem pada penelitian ini, dilakukan pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* guna memodelkan struktur sistem serta alur proses ekstraksi entitas lokasi secara sistematis. Berikut adalah diagram UML perancangan sistem chatbot:

4.2.1 Use Case Diagram



Gambar 4.1 Use Case Diagram Paimon Guide

Gambar 4.1 mempresentasikan rancangan *use case diagram* dari *chatbot* Paimon Guide yang menguraikan alur interaksi antara pengguna dengan fungsionalitas sistem. Pada arsitektur ini, aktor tunggal yang berperan adalah *User*, yang memiliki hak akses untuk mengeksekusi tiga skenario utama, yaitu mengakses menu *chatbot*, melakukan dialog dengan chatbot, dan mengelola profil akun. Interaksi operasional inti terjadi pada skenario dialog dengan *chatbot*, dimana *User* berinteraksi dengan

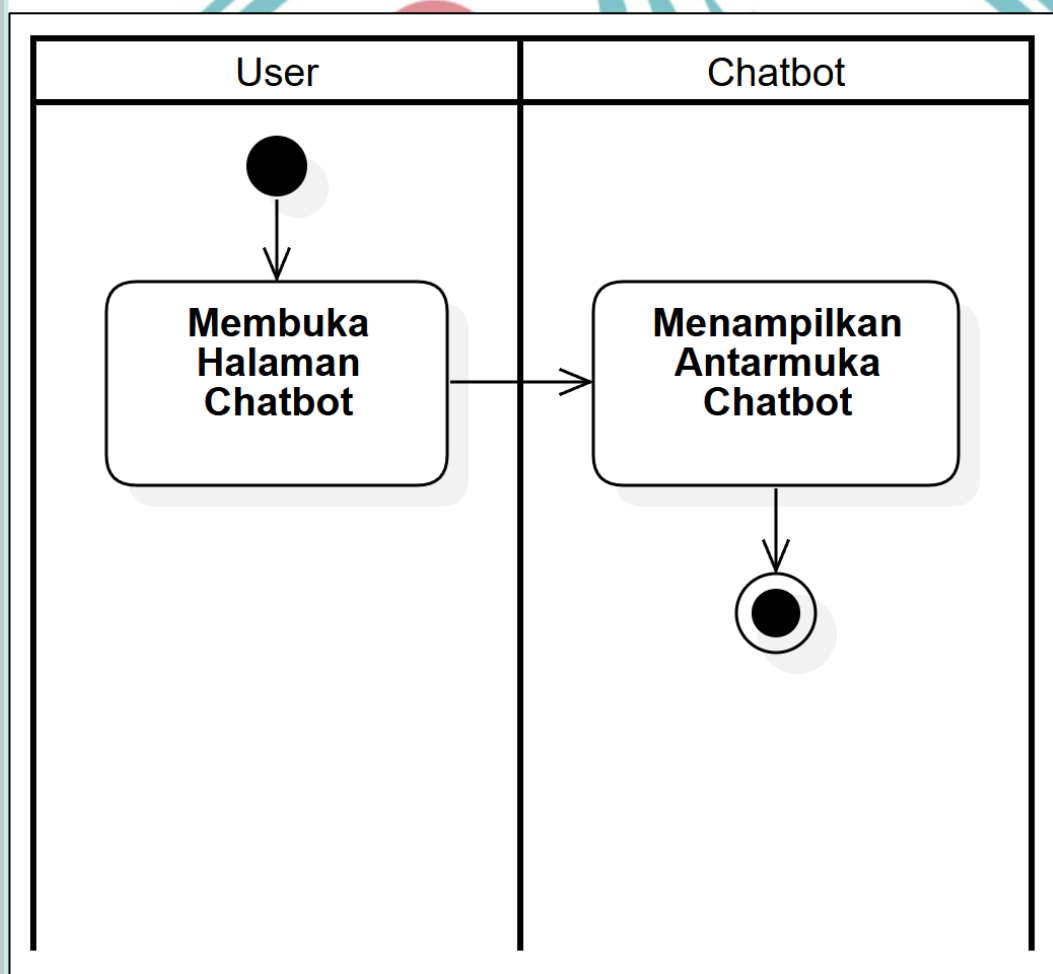


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

antarmuka *chatbot* untuk mencari panduan pembukaan area atau misi tertentu di dalam permainan Genshin Impact. Melalui fitur ini, masukan teks bahasa alami dari pengguna akan diproses oleh *back-end* sistem menggunakan metode *Named Entity Recognition* berbasis algoritma *Conditional Random Field*. Algoritma tersebut bekerja dengan mengekstrak entitas krusial seperti lokasi secara akurat, sehingga sistem mampu merespons dengan menyajikan informasi *quest* prasyarat secara instan.

4.2.2 Activity Diagram



Gambar 4.2 Activity Diagram Mengakses Menu Chatbot

Gambar 4.2 menggambarkan *activity diagram* ketika pengguna mengakses menu *chatbot*. Proses bermula pada *swimlane* aktor User yang ditandai dengan titik *start*, di mana pengguna melakukan aksi "Membuka Halaman Chatbot" dengan mengakses halaman utama sistem. Setelah permintaan akses diterima, alur proses

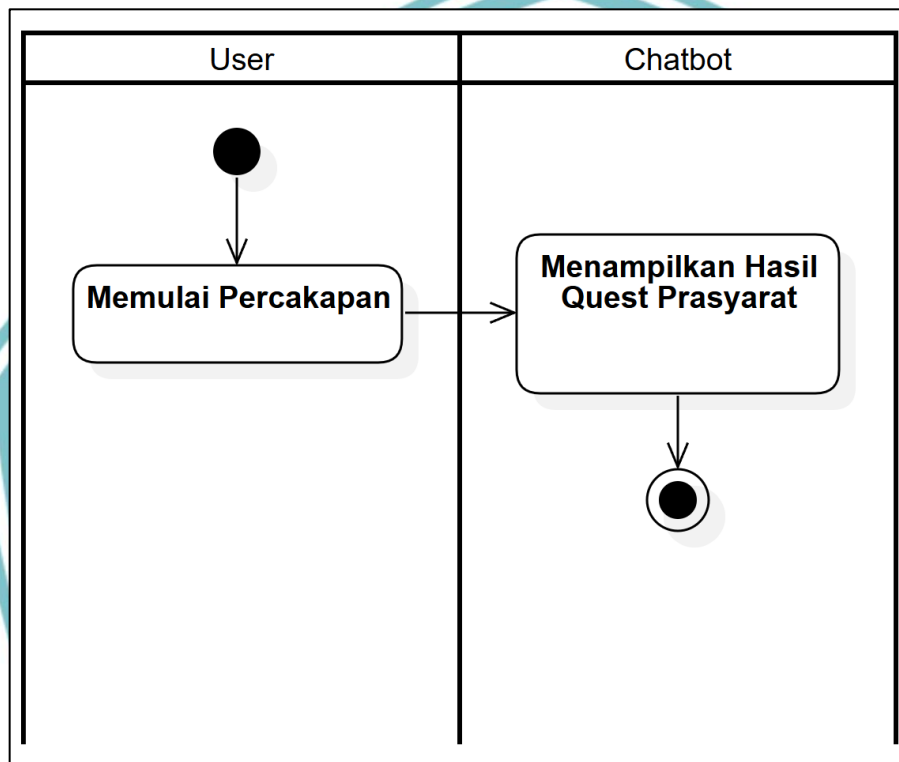


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

berpindah ke *swimlane* Chatbot. Sistem akan memproses permintaan tersebut dan merespons dengan mengeksekusi aksi "Menampilkan Antarmuka Chatbot" di layar perangkat. Setelah antarmuka dan seluruh elemen visual *chatbot* berhasil ditampilkan secara penuh kepada pengguna, alur aktivitas untuk skenario interaksi ini dinyatakan selesai, yang ditandai dengan titik END pada sisi Chatbot.



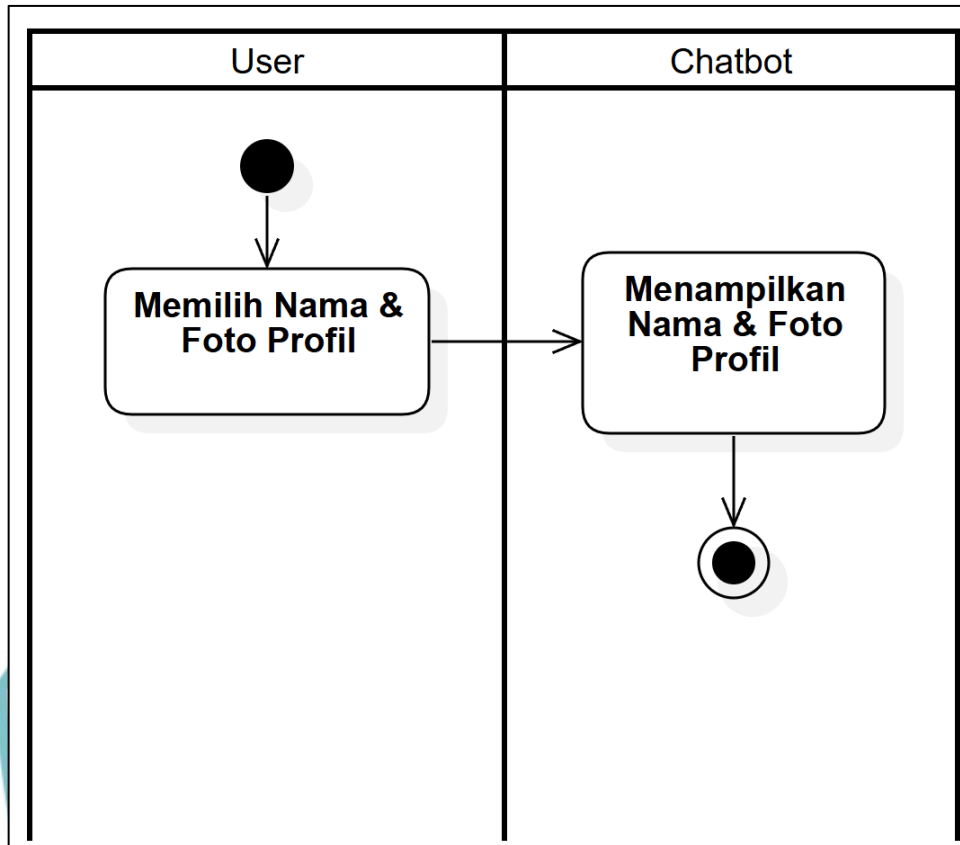
Gambar 4.3 Activity Diagram Melakukan Dialog Dengan Chatbot

Gambar 4.3 menggambarkan *activity diagram* ketika pengguna melakukan dialog di dalam menu *chatbot*. Proses bermula pada *swimlane* aktor User yang ditandai dengan titik *start*, di mana pengguna melakukan aksi "Memulai Percakapan" dengan menginputkan teks pertanyaan ke dalam sistem. Setelah masukan diterima, alur proses berpindah ke *swimlane* Chatbot. Sistem akan memproses pesan tersebut dan merespons dengan mengeksekusi aksi "Menampilkan Hasil Quest Prasyarat" di layar obrolan. Setelah informasi mengenai urutan atau prasyarat *quest* berhasil ditampilkan secara penuh kepada pengguna, alur aktivitas untuk skenario interaksi ini dinyatakan selesai, yang ditandai dengan titik *END* pada sisi Chatbot.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

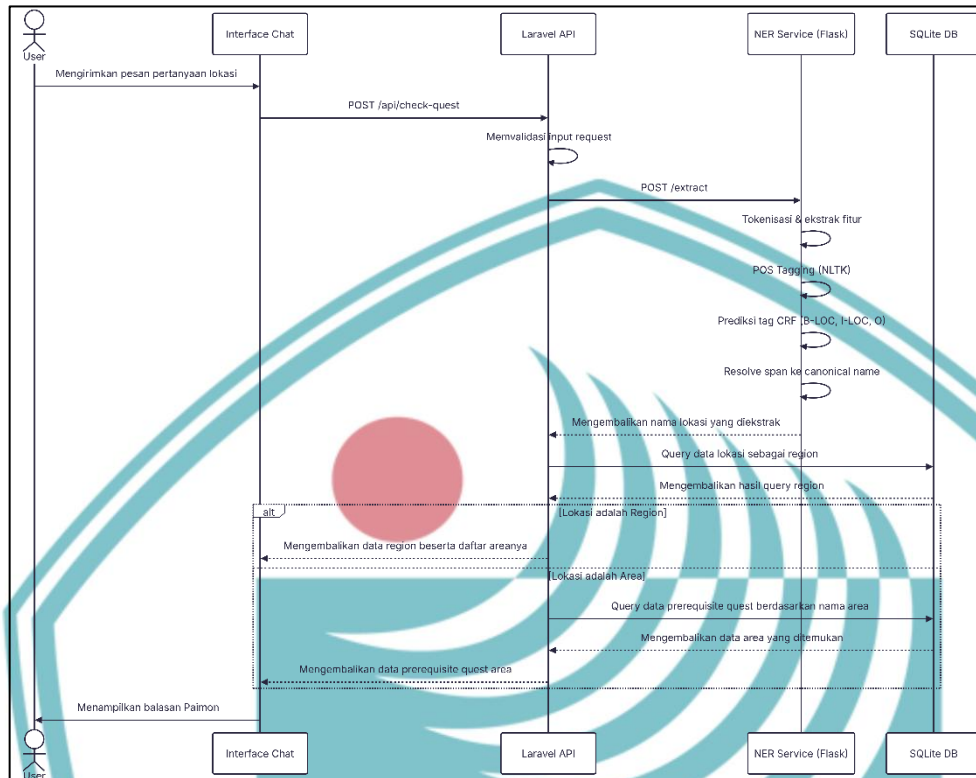


Gambar 4.4 Activity Diagram Mengelola Profil

Gambar 4.4 menunjukkan alur proses pada fitur mengelola profil di dalam sistem. Proses ini diawali pada *swimlane* aktor User, yang ditandai dengan simbol titik awal, di mana pengguna melakukan aktivitas "Memilih Nama & Foto Profil" untuk mengatur atau memperbarui identitas akun mereka. Setelah pengguna menentukan data yang diinginkan, alur berpindah menuju *swimlane* Chatbot. Sistem kemudian merespons input tersebut dengan menjalankan aksi "Menampilkan Nama & Foto Profil" pada antarmuka aplikasi. Setelah pembaruan data identitas pengguna berhasil diterapkan dan ditampilkan secara penuh, seluruh rangkaian proses pengelolaan profil ini dinyatakan selesai, yang direpresentasikan dengan simbol titik *END*.



4.2.3 Sequence Diagram



Gambar 4.5 Sequence Diagram Chatbot Paimon Guide

Gambar 4.5 menunjukkan *sequence diagram* yang mengilustrasikan urutan pengiriman dan penerimaan pesan antar objek secara waktu nyata ketika pengguna mencari informasi quest prasyarat. Pada diagram ini, terdapat satu aktor utama yaitu *User*, serta empat objek yang mengelola logika sistem: *Interface Chat*, *Laravel API*, *NER Service (Flask)*, dan *SQLite DB*.

Proses bermula saat *User* mengirimkan pesan pertanyaan lokasi melalui *Interface Chat*. *Interface Chat* kemudian meneruskan pesan tersebut ke *Laravel API* menggunakan metode *request POST /api/check-quest*. Setelah menerima data, *Laravel API* melakukan validasi *input request* secara internal sebelum mengirimkan *request POST /extract* menuju *back-end AI*, yaitu *NER Service (Flask)*.

Di dalam *NER Service (Flask)*, sistem menjalankan serangkaian pemrosesan kecerdasan buatan secara sekuensial. Proses internal ini meliputi tokenisasi dan ekstraksi fitur, *Part-of-Speech (POS) Tagging* memanfaatkan *NLTK*, memprediksi *tag* entitas lokasi menggunakan algoritma *CRF (B_LOC, I_LOC, O)*, serta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

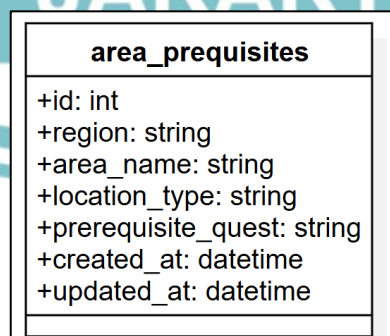
melakukan *resolve span* ke *canonical name* untuk membakukan nama lokasi. Setelah proses analisis bahasa alami ini selesai, NER Service (Flask) mengembalikan entitas nama lokasi yang berhasil diekstrak kepada Laravel API.

Laravel API mengeksekusi *query* ke SQLite DB untuk mencocokkan data lokasi tersebut sebagai *region* (wilayah utama), dan *database* mengembalikan hasil pencarian. Selanjutnya, sistem memasuki fragment *alt* yang membagi skenario respons berdasarkan hierarki lokasi yang dicari:

- Kondisi [Lokasi adalah *Region*]: Jika lokasi yang divalidasi merupakan *region*, Laravel API langsung mengembalikan data *region* beserta daftar area yang tergabung di dalamnya kepada Interface Chat.
- Kondisi [Lokasi adalah *Area*]: Jika lokasi merupakan suatu area spesifik, Laravel API akan kembali melakukan *query* ke SQLite DB untuk mencari data *prerequisite quest* berdasarkan nama *area*. Basis data kemudian mengembalikan data area yang relevan, dan Laravel API meneruskan informasi *prerequisite quest* tersebut ke Interface Chat.

Sebagai langkah terakhir dari keseluruhan sekuensi, Interface Chat menerima respons data dari API dan menampilkan balasan akhir *chatbot* (Paimon) kepada *User* di layar obrolan.

4.2.4 Class Diagram



Gambar 4.6 Class Diagram Database *area_prequisites*

Gambar 4.6 merupakan *class diagram* yang merepresentasikan struktur tabel *area_prequisites* pada basis data sistem untuk mengelola informasi lokasi permainan. Kelas ini memiliki atribut *id* dengan tipe data *integer* sebagai *primary*



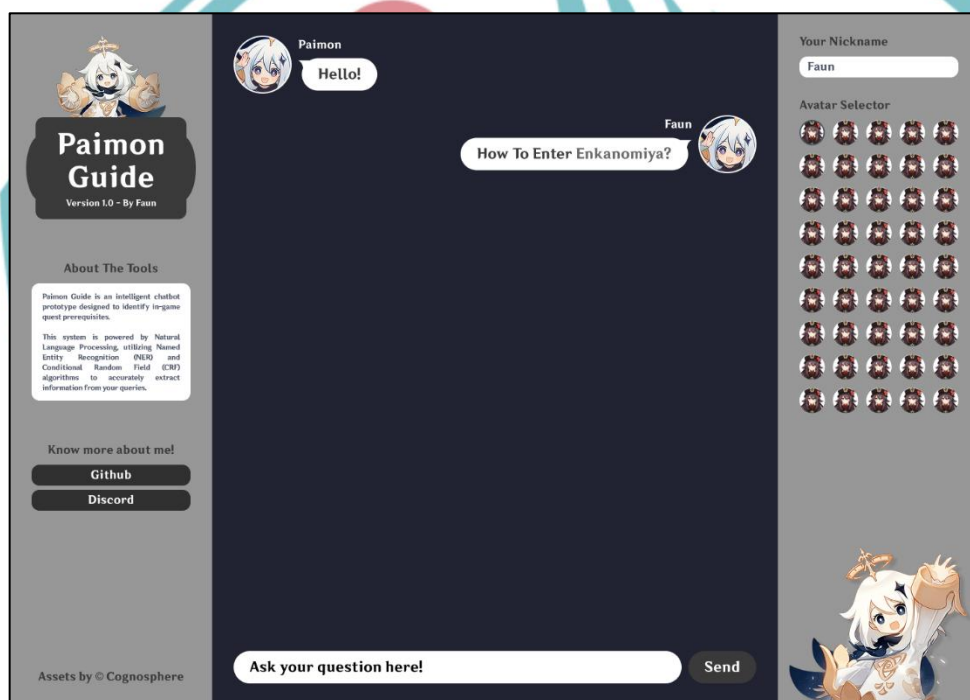
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

key, serta atribut bertipe *string* yang meliputi *region* untuk nama wilayah utama, *area_name* untuk nama zona spesifik, *location_type* untuk kategori lokasi, dan *prerequisite_quest* untuk menyimpan nama misi prasyaratnya. Selain itu, terdapat atribut *created_at* dan *updated_at* bertipe *datetime* yang berfungsi mencatat waktu pembuatan dan pembaruan riwayat data. Struktur kelas statis ini menjadi landasan operasional bagi sistem dalam mengeksekusi *query* pencarian data prasyarat berdasarkan entitas lokasi yang berhasil diekstrak oleh *chatbot*.

4.2.5 Wireframe



Gambar 4.7 Wireframe Paimon Guide

Gambar 4.7 menampilkan wireframe dari sistem *chatbot* Paimon Guide. Tampilan halaman ini secara visual terbagi ke dalam tiga kolom segmen utama. Panel sebelah kiri memuat sekumpulan elemen statis yang terdiri dari logo aplikasi, *text box* untuk deskripsi sistem, serta dua *action buttons*. Bagian tengah dikhususkan sebagai *chat area* yang menampilkan elemen *chat bubbles*. Pada sisi bawah *chat area* terdapat *input field* berbentuk memanjang yang berdampingan dengan tombol pengiriman pesan. Panel sebelah kanan memuat elemen *form* yang terdiri dari *text box* untuk penamaan dan sebuah matriks yang berisi sekumpulan ikon gambar melingkar sebagai opsi pemilihan avatar pengguna.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3 Implementasi Model

4.3.1 Business Understanding

Tahap *business understanding* berfokus pada identifikasi kendala utama yang dialami oleh pemain permainan Genshin Impact, yaitu kurang efisiennya proses pencarian informasi secara manual mengenai *quest* prasyarat untuk membuka akses ke suatu wilayah atau area tertentu. Diperlukan pemahaman mendalam mengenai karakteristik bahasa alami pengguna, termasuk penggunaan variasi penyebutan, singkatan tidak standar, serta istilah informal yang sering digunakan pemain saat bertanya di komunitas daring. Pemahaman terhadap konteks masalah dan bahasa ini krusial sebagai dasar untuk merancang sistem *chatbot* yang mampu mengekstrak entitas lokasi secara tepat.

4.3.2 Data Understanding

Tahap pemahaman data (*data understanding*) mengacu pada proses pengumpulan dan pengenalan terhadap data yang akan digunakan untuk melatih model algoritma kecerdasan buatan. Tahap ini dimaksudkan untuk memahami karakteristik dan sumber data mentah yang telah dikumpulkan. Berikut adalah data yang akan dieksplorasi dan digunakan dalam pelatihan model:

- Data riwayat percakapan dari *server* Discord resmi Genshin Impact pada rentang tahun 2021 hingga 2025, yang secara spesifik telah difilter menggunakan teknik *purposive sampling* untuk mendapatkan teks pertanyaan mengenai cara mengakses suatu area.
- Data terminologi permainan yang bersumber dari dokumentasi Genshin Impact Wiki, yang memuat daftar entitas penting berupa nama lokasi dan kompilasi *quest* prasyarat.

Dataset utama yang digunakan terdiri dari 21 kalimat. Kalimat-kalimat tersebut telah dianotasi dan divalidasi ke dalam 3 kelas entitas (O, B-LOC, I-LOC) yang merujuk secara ketat pada terminologi penamaan area resmi di dalam permainan Genshin Impact.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 2. Deskripsi Label pada Dataset

Label	Deskripsi	Kode Label
O	Token yang bukan entitas	0
B-LOC	Awal dari entitas lokasi	1
I-LOC	Bagian dalam entitas lokasi	2

Contoh isi 1 kalimat dataset untuk kalimat “How to enter chasm the underground mines”.

Tabel 3. Contoh Kalimat pada Dataset

Token	How	to	enter	chasm	the	underground	mines
Label	O	O	O	B-LOC	I-LOC	I-LOC	I-LOC

4.3.3 Data Preparation

Tahap persiapan data (*data preparation*) adalah sebuah tahap setelah *data understanding*, di mana data teks percakapan pemain yang masih bersifat tidak terstruktur akan diolah agar siap digunakan dalam proses pemodelan. Pada tahap ini, data teks mentah diproses melalui tahapan *preprocessing* yang terdiri dari tokenisasi, penghapusan karakter khusus, dan pembersihan tanda baca dengan tetap mempertahankan struktur singkatan penting (seperti awalan 'Mt.'). Teks yang telah dinormalisasi kemudian diselaraskan untuk kebutuhan ekstraksi fitur.

Selain itu, pada tahap ini juga dilakukan proses augmentasi data dengan teknik *entity replacement* dari sumber data Wiki Genshin Impact. Hal ini bertujuan untuk menambah variasi data latih terkait penggunaan singkatan atau istilah informal, sehingga data yang digunakan menjadi lebih kaya dan menghasilkan model yang memiliki performa lebih baik.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
import json
import pickle
import random
import re

import matplotlib.pyplot as plt
import nltk
from sklearn.metrics import ConfusionMatrixDisplay, accuracy_score, confusion_matrix, classification_report
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
```

Gambar 4.8 Import Library Pendukung

Hal pertama yang harus dilakukan dalam proses *data preparation* adalah melakukan impor pada *library* yang diperlukan. Masing-masing *library* mempunyai fungsinya tersendiri dalam pemrosesan *Natural Language Processing* (NLP). *Library* pertama yang diimpor adalah json dan pickle untuk membaca struktur data dari eksternal dan menyimpan model. Selanjutnya diimpor *library* random yang berfungsi untuk melakukan proses pengacakan saat tahap augmentasi kalimat. Setelahnya diimpor library re (Regex) yang digunakan untuk mendeteksi pola teks. Diimpor pula *library* nltk (*Natural Language Toolkit*) yang krusial untuk menganalisis *Part of Speech* (POS). Terakhir, diimpor modul dari scikit-learn yang مخصوص untuk pembagian *dataset* dan evaluasi metrik performa, serta matplotlib untuk kebutuhan visualisasi matriks evaluasi.

```
LABELS = ["O", "B-LOC", "I-LOC"]
LOCATION_CONNECTORS = {"of", "de", "al", "al-", "the"}
ALIAS_MAP = {
    "mondo": "mondstadt",
    "mondstat": "mondstadt", # Typo yang sangat sering terjadi
    "dragon spine": "dragonspine",
    "enka": "enkanomiya",
    "chasm underground": "the chasm: underground mines",
    # ... (Singkatan lainnya untuk Sumeru, Fontaine, Natlan, dst)
}
_ABBREV_RE = re.compile(r"^[A-Z][a-z]{0,3}\.?$")
```

Gambar 4.9 Inisialisasi Variabel Global dan Mapping Alias

Setelah impor selesai, didefinisikan berbagai variabel global. Pertama dibuat variabel LABELS yang menetapkan format IOB ("O", "B-LOC", "I-LOC") sebagai landasan utama algoritma CRF. Selanjutnya dibuat variabel ALIAS_MAP, yang mana di dalam variabel ini berisi *dictionary* dari istilah informal pemain (seperti



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

"mondo", "enka") ke dalam *canonical name* di permainan Genshin Impact. Pemetaan ini krusial sebagai fondasi agar mesin algoritma dapat menyamakan konteks bahasa komunitas dengan dokumentasi permainan.

```
def load_gazetteer(json_path: str) -> tuple[set[str], set[str]]:
    with open(json_path, "r", encoding="utf-8") as handle:
        data = json.load(handle)

    names = flatten_gazetteer(data)
    for expansion in ALIAS_MAP.values():
        names.add(normalize_name(expansion))

    return names, build_token_set(names)

def flatten_gazetteer(data: dict) -> set[str]:
    names: set[str] = set()
    for region_name, region_data in data.items():
        names.add(normalize_name(region_name))
        if not isinstance(region_data, dict):
            continue
        for area_name, area_data in region_data.get("areas", {}).items():
            names.add(normalize_name(area_name))
            if isinstance(area_data, dict):
                for sub_area in area_data.get("sub_areas", []):
                    cleaned = normalize_name(sub_area)
                    if cleaned:
                        names.add(cleaned)
        names.discard("")
    return names

def build_token_set(names: set[str]) -> set[str]:
    tokens: set[str] = set()
    for name in names:
        for word in name.split():
            cleaned = word.strip('.,;":\'(){}')
            if cleaned and len(cleaned) > 1:
                tokens.add(cleaned)
    return tokens
```

Gambar 4.10 Memuat dan Merapihkan Data Terminologi

Setelah variabel siap, sistem mengenali data nama lokasi resmi melalui berkas eksternal menggunakan fungsi `load_gazetteer`. Di dalam fungsi ini, file `genshin_areas.json` dibuka dan isinya diratakan menggunakan fungsi `flatten_gazetteer`. Data JSON dengan hierarki Region, Area, dan Sub-Area diekstraksi ke dalam himpunan dataset. Setiap lokasi diubah menjadi *lowercase* dan dibersihkan dari simbol. Kumpulan nama lokasi utuh ini juga dipecah kembali menjadi *token set* melalui fungsi `build_token_set`. Kumpulan data ini memastikan integritas nama resmi yang dibaca algoritma tetap terjaga.

```
Gazetteer sebelum flattening:
{'Mondstadt': {'type': 'Region', 'sample_areas': ['Brightcrown Mountains', 'Galesong Hill', 'Starfell Valley']}}

Flattened gazetteer sample (5 nama pertama):
['a lonely place', 'a very bright place', 'aaru village', 'abdu pit']

Flattened gazetteer token sample (5 token pertama):
['aaru', 'abdu', 'academy', 'ad-hoc', 'adeptus-s']
```

Gambar 4.11 Hasil Flattening Gazetteer



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
def tokenize_locations(text: str) -> list[str]:
    for old, new in (("u201c", "'"), ("u201d", "'"), ("u2018", "'"), ("u2019", "'")):
        text = text.replace(old, new)

    tokens: list[str] = []
    for token in text.split():
        while len(token) > 1 and token[0] == "'" and token[-1] == "'":
            token = token[1:-1]
            token = token.strip("'")

        if token.startswith("(") and token.endswith(")"):
            continue
        token = re.sub(r"\"([^\"]*)\"", "", token)

        if not _ABBREV_RE.match(token):
            token = token.rstrip("?!.,;")

        if token:
            tokens.append(token)

    merged_tokens: list[str] = []
    idx = 0
    while idx < len(tokens):
        if idx + 1 < len(tokens) and tokens[idx] == "Mt." and tokens[idx + 1] and tokens[idx + 1][0].isalpha():
            merged_tokens.append(f"{tokens[idx]} {tokens[idx + 1]}")
            idx += 2
            continue
        merged_tokens.append(tokens[idx])
        idx += 1

    return merged_tokens

def get_pos_tags(tokens: list[str]) -> list[str]:
    ensure_pos_tagger()
    return [tag for _, tag in nltk.pos_tag(tokens)]

def normalize_token(token: str) -> str:
    return normalize_name(token).strip('.,;!?')
```

Gambar 4.12 Proses Tokenisasi Teks, Normalisasi, dan POS Tagging

Data percakapan pemain memiliki format teks acak yang harus dirapikan menggunakan fungsi `tokenize_locations`. Teks dipecah menjadi array per kata (*token*), dengan menyeleksi tanda baca yang tidak penting dan menggabungkan singkatan penamaan lokasi spesifik seperti "Mt." (Gunung). Sebelum token diproses lebih jauh, fungsi `normalize_token` dilibatkan untuk membersihkan sisa tanda baca di pinggir kata dan menyeragamkan bentuknya menjadi huruf kecil agar sesuai dengan data pada *gazetteer*. Setelah terpecah dan ternormalisasi dengan sempurna, array *token* dikirim ke fungsi `get_pos_tags`. Fungsi ini memanfaatkan *library* `nltk` untuk mendeteksi peran gramatikal kata tersebut, seperti kata benda maupun tata bahasa dasar lainnya secara otomatis, yang mana akan menambah akurasi pada pengenalan entitas.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
Tokenization before:
How to enter "The Chasm: Underground Mines?"

Tokenization after:
['How', 'to', 'enter', 'The', 'Chasm:', 'Underground', 'Mines']

Normalized token preview:
[('How', 'how'), ('to', 'to'), ('enter', 'enter'), ('The', 'the'), ('Chasm:', 'chasm'), ('Underground', 'underground'), ('Mines', 'mines')]
```

Gambar 4.13 Contoh Hasil Tokenization dan Normalization

Model CRF tidak membaca kata mentah secara langsung, sehingga data harus diubah menjadi serangkaian parameter bobot linguistik. Fungsi utama yang bertugas untuk itu adalah `word2features` yang diaktifkan dengan metode iterasi lewat fungsi `sent2features`. *Dictionary* akan diisi oleh sekumpulan kondisi fitur seperti format awalan, penanda awal/akhir kalimat, integrasi data gazetteer, serta nilai komparasi untuk kata sebelum (-1) dan sesudah (+1) token terkait. Struktur *dictionary* yang komprehensif ini merepresentasikan keterkaitan tata bahasa struktural di sekitar entitas lokasi

Gambar 4.14 Ekstraksi Fitur dalam Feature Engineering

Nama Fitur	Tipe Data	Penjelasan
bias	Float	Nilai dasar/konstanta pembobotan awal
word.isdigit()	Boolean	Deteksi apakah token sepenuhnya berupa angka
word.prefix2	String	Ekstraksi 2 karakter pertama dari kata
word.prefix3	String	Ekstraksi 3 karakter pertama dari kata
word.suffix2	String	Ekstraksi 2 karakter terakhir dari kata
word.suffix3	String	Ekstraksi 3 karakter terakhir dari kata
postag	String	Tag POS utuh dari kata saat ini
postag[:2]	String	Generalisasi awalan POS Tag (2 karakter pertama)
is_sentence_start	Boolean	Tanda token berada di posisi awal kalimat
is_sentence_end	Boolean	Tanda token berada di posisi akhir kalimat



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

is_location_connector	Boolean	Deteksi token berupa preposisi penunjuk lokasi
is_in_genshin_map	Boolean	Validasi kecocokan utuh dengan <i>Gazetteer</i>
gazetteer_word_overlap	Boolean	Status kemiripan per-token dengan <i>Gazetteer</i>
gazetteer_phrase_overlap	Boolean	Deteksi irisan kata dengan frasa utuh di kalimat
word.trigram: {trigram}	Boolean	Ekstraksi 3 karakter beruntun (<i>trigram</i>) dari sub-kata
BOS	String	Deteksi apakah kalimat awal
EOS	String	Deteksi apakah kalimat akhir
±1:postag	String	Tag POS dari kata sebelum/sesudahnya
±1:postag[:2]	String	Generalisasi awalan POS Tag dari kata sebelum/sesudahnya
±1:is_connector	Boolean	Deteksi kata hubung tempat pada kata sebelum/sesudahnya
±1:gazetteer_word_overlap	Boolean	Deteksi kecocokan kamus pada kata sebelum/sesudahnya

Langkah krusial berikutnya adalah memproses data latih manual yang telah disiapkan sebelumnya dengan memanggil fungsi `parse_tsv_dataset`. Kode pada bagian ini mengeksekusi proses pembacaan berkas dataset TSV baris demi baris menggunakan `for loop`.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
def parse_tsv_dataset(dataset_path: str) -> list[tuple[list[str], list[str]]]:
    sentences = []
    tokens = []
    tags = []
    skipped_lines = []

    with open(dataset_path, "r", encoding="utf-8") as handle:
        for line_number, raw_line in enumerate(handle, start=1):
            line = raw_line.strip()
            if not line:
                if tokens:
                    sentences.append((tokens, tags))
                    tokens, tags = [], []
                continue

            if "\t" in line:
                parts = line.split("\t", 1)
            else:
                parts = line.rsplit(None, 1)

            if len(parts) != 2:
                skipped_lines.append((line_number, line))
                continue

            token, tag = parts[0].strip(), parts[1].strip()
            if tag not in {"O", "B-LOC", "I-LOC"}:
                skipped_lines.append((line_number, line))
                continue

            tokens.append(token)
            tags.append(tag)

        if tokens:
            sentences.append((tokens, tags))

        if skipped_lines:
            print("Skipped malformed dataset lines:", skipped_lines[:5])

    return sentences
```

Gambar 4.15 Parsing Dataset Berformat IOB

Setiap baris dipisahkan dengan delimiter dan dibagi menjadi struktur *token* dan *tag*. Kode tersebut memastikan bahwa label yang masuk secara konsisten tervalidasi menggunakan aturan himpunan "O", "B-LOC", atau "I-LOC". Ketika sebuah baris kosong dicapai, sistem mengidentifikasi bahwa satu kalimat utuh telah berakhir, lalu membungkus array kata-kata tersebut sebagai sebuah pasangan kalimat berannotasi lengkap.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
def extract_first_location_span(tags: list[str]) -> tuple[int, int] | None:
    start_index = None
    end_index = None

    for index, tag in enumerate(tags):
        if tag == "B-LOC":
            if start_index is not None:
                break
            start_index = index
            end_index = index + 1
        elif tag == "I-LOC" and start_index is not None:
            end_index = index + 1
        elif start_index is not None:
            break

    if start_index is None or end_index is None:
        return None
    return start_index, end_index

def build_augmented_sentence(template_tokens: list[str], template_tags: list[str], location_name: str) -> tuple[list[str], list[str]] | None:
    location_span = extract_first_location_span(template_tags)
    if location_span is None:
        return None

    location_tokens = tokenize_locations(location_name)
    if not location_tokens:
        return None

    start_index, end_index = location_span
    augmented_tokens = template_tokens[:start_index] + location_tokens + template_tokens[end_index:]
    augmented_tags = template_tags[:start_index] + ["B-LOC"] + ["I-LOC"] * (len(location_tokens) - 1) + template_tags[end_index:]
    return augmented_tokens, augmented_tags

def augment_with_gazetteer(
    sentences,
    gazetteer_names,
    max_examples: int = 10000,
    random_state: int = 42,
    templates_per_area: int = 3,
```

Gambar 4.16 Proses Augmentasi Data (Entity Replacement)

Tahap terakhir dalam melatih data adalah teknik augmentasi *Entity Replacement*. Pertama, letak entitas lokasi pada kalimat template akan diekstrak menggunakan fungsi `extract_first_location_span`. Koordinat indeks awal ("B-LOC") dan indeks akhir ("I-LOC") yang dihasilkan lalu digunakan oleh fungsi `build_augmented_sentence` untuk memotong dan melepaskan porsi lokasi yang lama. Setelah itu, susunan array kalimat dijahit kembali dengan menyelipkan token lokasi yang baru. Susunan tag IOB dikonfigurasi ulang secara otomatis pada saat perakitan array tersebut. Proses ini dijalankan melalui nested loop di dalam `augment_with_gazetteer` yang membongkar nama kandidat dan template secara acak untuk mencegah duplikasi data latih yang dihasilkan. Proses ini dibatasi oleh `templates_per_area` yang diatur secara sebanyak 3.

```
Dataset sebelum augmentation: 21
Dataset setelah augmentation: 1032
Dataset setelah augmentation (Preview 1 Location):
3 Kalimat augmentation terakhir:
([['how', 'to', 'enter', 'vourukasha', 'oasis'], ['0', '0', '0', 'B-LOC', 'I-LOC']])
([['what', 'are', 'the', 'requirements', 'to', 'access', 'vourukasha', 'oasis', 'on', 'release'], ['0', '0', '0', '0', '0', '0', 'B-LOC', 'I-LOC', '0', '0']])
([['I', 'just', 'wanna', 'know', 'how', 'to', 'enter', 'vourukasha', 'oasis'], ['0', '0', '0', '0', '0', '0', '0', 'B-LOC', 'I-LOC']])

Total kalimat (asli + augmentasi): 1053
Kalimat unik: 1053
Duplikat: 0

Tidak ada kalimat duplikat - augmentasi bersih.
```

Gambar 4.17 Hasil Augmentasi Dataset



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3.4 Modeling

Setelah dataset melalui tahap ekstraksi fitur dan augmentasi, matriks fitur linguistik tersebut diproses lebih lanjut menggunakan algoritma *Conditional Random Field* (CRF). Pelatihan CRF berpusat pada optimasi bobot statistik terhadap fitur-fitur teks. Langkah pertama yang dilakukan mesin saat menerima data kalimat adalah menghitung besaran skor relasi antar kata dengan label serta bobot relasi antar label yang saling bersebelahan.

```
def _get_potentials(self, x, W_node, W_trans):
    # Menghitung probabilitas kondisional p(y|x) dari label y terhadap data masukan x
    # W_node bobot fitur node (lamda k)
    # Kalau W_trans buat bobot fitur edge (lambda k')
    N = len(x)
    node_score = np.zeros((N, self.num_labels))
    for t in range(N):
        for feat, val in x[t].items():
            if feat in self.feature_to_idx:
                feat_idx = self.feature_to_idx[feat]
                # Rumus Node: Sum k lambda k * fk (yt,x,t)
                node_score[t, :] += W_node[feat_idx, :] * val

    # Di logsumexp biar pas di lakuin exponen nilainya ga gede
    node_shift = np.max(node_score, axis=1, keepdims=True)
    # Rumus Node Potential: phi t (yt, x) = exp (Sum k lamda k fk)
    node_potential = np.exp(node_score - node_shift)

    # Di logsumexp biar pas di lakuin exponen nilainya ga gede
    edge_shift = np.max(W_trans)
    # Rumus Edge Potential : psit(yt, yt+1, x) = exp(sum k' lambda k' fk'(yt, yt+1, x, t))
    # Diabaikan token masukan (x) dan waktu (t) biar hanya menghitung transisi antar label
    # Fitur fk' diasumsikan konstan (=1) untuk setiap pasangan label yang valid
    # Sehingga persamaannya disederhanakan murni menjadi: exp(lambda k')
    edge_potential = np.exp(W_trans - edge_shift)

    return node_potential, edge_potential, node_shift.flatten(), edge_shift
```

Gambar 4.18 Fungsi Perhitungan *Node Potential* & *Edge Potential*

Berdasarkan Persamaan (2.3) untuk perhitungan *node potential* (*node_potential*) dan Persamaan (2.4) untuk perhitungan *edge potential* (*edge_potential*), fase ini melakukan kalkulasi matriks fitur awal. Pada perulangan *for*, operasi penambahan (*+=*) merepresentasikan operator sigma ($\sum$) yang mengalikan bobot (*W_node*) dengan fitur *boolean* bernilai satu atau nol (*val*). Setelah skor fitur tersimpan di *node_score* dan bobot transisi di *W_trans*, kode menggunakan fungsi eksponensial (*np.exp*). Pengurangan dengan nilai maksimum melalui variabel *node_shift* dan *edge_shift* adalah implementasi dari metode *logsumexp*. Modifikasi ini digunakan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

untuk mencegah terjadinya *overflow*, yaitu ketika operasi perpangkatan menghasilkan nilai yang melampaui batas kapasitas memori.

```
def _forward_backward(self, node_potential, edge_potential):
    N = node_potential.shape[0]
    Y = self.num_labels

    alpha = np.zeros((N, Y))
    k = np.zeros(N) # Array untuk menyimpan Scaling Factor (kt)

    # Inisialisasi nilai forward pass pada urutan pertama (t=1)
    # alpha 1 [y1] = phi 1 (y1,x)
    alpha[0] = node_potential[0]
    k[0] = np.sum(alpha[0])
    # normalisasi untuk t=1
    alpha[0] /= k[0] if k[0] != 0 else 1e-100

    for t in range(1, N):
        # Menghitung nilai forward pass pada urutan ke-t (t >= 2) dengan scaling factor kt
        # Rumus: alpha t[yt] = kt * sum yt-1 alpha t-1 [yt-1] * phi t(yt, x) * psi t-1 (yt-1, yt, x))
        alpha[t] = np.dot(alpha[t-1], edge_potential) * node_potential[t]
        k_t = np.sum(alpha[t])
        # Menerapkan pembagian dengan scaling factor kt
        alpha[t] /= k_t if k_t != 0 else 1e-100
        k[t] = k_t

    beta = np.zeros((N, Y))
    mu = np.zeros(N) # Array untuk menyimpan scaling factor mu t
```

Gambar 4.19 Fungsi Perhitungan *Forward Pass*

Melihat Persamaan (2.8), inisialisasi pada elemen indeks ke-0 ($\alpha[0]$) merupakan ketetapan batas probabilitas awal kalimat. Lalu berdasarkan Persamaan (2.9), perulangan *for* menghitung *forward pass* untuk waktu $t \geq 2$. Fungsi perkalian matriks `np.dot` mengalikan nilai *forward pass* sebelumnya ($\alpha[t-1]$) dengan nilai `edge_potential` dan `node_potential` saat ini. Variabel `k_t` digunakan untuk menormalisasi nilai probabilitas di setiap iterasi agar perkalian nilai desimal tidak berujung menjadi angka nol mutlak (*underflow*) pada kalimat masukan yang panjang.

```
beta = np.zeros((N, Y))
mu = np.zeros(N) # Array untuk menyimpan scaling factor mu t

# Inisialisasi nilai backward pass pada urutan terakhir (t=T)
# Rumus: betaT[yT] = 1 / S (Di sini disederhanakan menjadi 1.0 sebelum scaling)
beta[N-1] = 1.0
mu_t = np.sum(beta[N-1])
# Normalisasi untuk t=T
beta[N-1] /= mu_t if mu_t != 0 else 1e-100
mu[N-1] = mu_t

for t in range(N-2, -1, -1):
    # Menghitung nilai backward pass pada urutan ke-t (t < T) dengan scaling factor mu t
    # Rumus: beta t[yt] = mu t * sum yt+1 (beta t+1 [yt+1] * phi t+1(yt+1, x) * psit(yt, yt+1, x))
    beta[t] = np.dot(edge_potential, beta[t+1] * node_potential[t+1])
    mu_t = np.sum(beta[t])
    # Menerapkan pembagian dengan scaling factor mu t
    beta[t] /= mu_t if mu_t != 0 else 1e-100
    mu[t] = mu_t
```

Gambar 4.20 Fungsi Perhitungan *Backward Pass*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Dari Persamaan (2.10), tahap ini diawali dengan menginisialisasi indeks paling ujung ($\beta[N-1]$) pada nilai konstan. Lalu melihat Persamaan (2.11), blok perulangan beroperasi mundur dari $N-2$ hingga 0 untuk mengakumulasi *backward pass*. Operasi np.dot digunakan untuk menarik probabilitas gabungan dari masa depan ($\beta[t+1] * \text{node_potential}[t+1]$) menuju masa kini ($\beta[t]$). Sama seperti operasi sebelumnya, variabel pembagi independen μ_t digunakan agar perhitungan matriks terhindar dari permasalahan *underflow*.

```
# Menghitung fungsi normalisasi Z(x) dari distribusi probabilitas kondisional
# Rumus log Z(x) dihitung dengan mengakumulasikan nilai log dari seluruh scaling factor
Z_x = np.sum(np.log(k))
```

Gambar 4.21 Perhitungan Fungsi Normalisasi

Berdasarkan Persamaan (2.2), bagian ini memecahkan komputasi pencarian fungsi normalisasi. Fungsi gabungan $\text{np.sum}(\text{np.log}(k))$ menghitung nilai probabilitas total $Z(x)$ dengan cara mengakumulasikan nilai logaritma dari seluruh variabel *scale factor forward pass* (k).

```
def _marginal_probabilities(self, alpha, beta, node_potential, edge_potential):
    N = alpha.shape[0]
    Y = self.num_labels

    node_marginals = np.zeros((N, Y))
    edge_marginals = np.zeros((N-1, Y, Y))

    for t in range(N):
        # Menghitung probabilitas node yakni peluang kemunculan label kelas kata yt
        # Rumus: P(yt|x) = (alpha t[yt] * beta t[yt]) / Z(x)
        # Pembagian Z(x) ditangani oleh sum_node karena alpha dan beta sudah di-scale
        node_marginals[t] = alpha[t] * beta[t]
        sum_node = np.sum(node_marginals[t])
        if sum_node > 0: node_marginals[t] /= sum_node

    for t in range(N - 1):
        # Menghitung probabilitas edge yakni peluang transisi dari label yt ke yt+1.
        # Rumus: P(yt, yt+1|x) = (alpha t * psi * phi t+1 * beta t+1) / Z(x)
        edge_marg = np.outer(alpha[t], beta[t+1] * node_potential[t+1]) * edge_potential
        sum_edge = np.sum(edge_marg)
        if sum_edge > 0: edge_marg /= sum_edge
        edge_marginals[t] = edge_marg

    return node_marginals, edge_marginals
```

Gambar 4.22 Perhitungan Probabilitas Marginal Node & Edge



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Melihat Persamaan (2.12), perhitungan probabilitas *node* pada (*node_marginals[t]*) dilakukan dengan mengalikan matriks *forward pass* ($\alpha[t]$) dengan matriks *backward pass* ($\beta[t]$). Variabel pembagi *sum_node* menormalisasi besaran nilai agar total probabilitas bernilai persis 1. Lalu dengan Persamaan (2.13), probabilitas *edge* dihitung menggunakan fungsi perkalian produk luar matriks (*np.outer*). Fungsi ini mengalikan matriks *forward pass* saat ini ($\alpha[t]$) dengan transisi *backward pass* pada masa depan ($\beta[t+1] * \text{node_potential}[t+1]$).

```
def gradient(self, weights, proc_X, y_idx):
    # Ekstrak matriks bobot lambda k dan lambda k'
    W_node = weights[:self.num_features * self.num_labels].reshape((self.num_features, self.num_labels))
    W_trans = weights[self.num_features * self.num_labels:].reshape((self.num_labels, self.num_labels))

    # Inisialisasi matriks gradien Gk dan G k'
    grad_node = np.zeros_like(W_node)
    grad_trans = np.zeros_like(W_trans)

    # Inisialisasi nilai Log-Likelihood
    log_likelihood = 0.0

    for i in range(len(proc_X)):
        x_seq = proc_X[i]
        y_seq = y_idx[i]
        N = len(x_seq)

        node_potential, edge_potential, node_shift, edge_shift = self._get_potentials(x_seq, W_node, W_trans)
        alpha, beta, k, mu, Z_x_scaled = self._forward_backward(node_potential, edge_potential)
        # Mengembalikan fungsi normalisasi Z(x) ke skala asli dari logsumexp
        Z_x = Z_x_scaled + np.sum(node_shift) + (N - 1) * edge_shift
        node_marginals, edge_marginals = self._marginal_probabilities(alpha, beta, node_potential, edge_potential)
```

Gambar 4.23 Inisialisasi Parameter Untuk Menghitung Gradien

Mencari *loss function* untuk menyesuaikan nilai gradien parameter bobot, agar dapat memproduksi rasio prediksi yang lebih akurat pada perulangan berikutnya, dipanggil fungsi *_get_potentials* dan *_forward_backward*. Eksekusi fungsi *forward-backward* akan mengembalikan nilai *Z_x*. Lalu *_marginal_probabilities* untuk menghasilkan probabilitas lokal berupa tebakan peluang sementara dari mesin. Hasil kalkulasi dari Persamaan (2.12) dan (2.13) ini kemudian ditampung di dalam *array* matriks variabel *node_marginals* dan *edge_marginals*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
# Rumus Empiris:  $\sum t=1^T \sum k \lambda k f_k(y_{t-1}, y_t, x, t)$ 
seq_score = 0.0
for t in range(N):
    y_t = y_seq[t]
    for feat, val in x_seq[t].items():
        if feat in self.feature_to_idx:
            feat_id = self.feature_to_idx[feat]
            seq_score += W_node[feat_id, y_t] * val
            # Nilai observasi empiris untuk gradien Node
            grad_node[feat_id, y_t] -= val

    if t > 0:
        y_prev = y_seq[t-1]
        seq_score += W_trans[y_prev, y_t]
        # Nilai observasi empiris untuk gradien Edge
        grad_trans[y_prev, y_t] -= 1.0
```

Gambar 4.24 Perhitungan Fitur Empiris Untuk *Node* dan *Edge*

Dari Persamaan (2.5) melalui perulangan for, skor fitur empiris (*node* dan *edge*) akan dikalikan dengan bobot aktual (W_{node} dan W_{trans}) untuk diakumulasikan ke variabel seq_score . Pada blok kondisional $t > 0$, sistem mengeksekusi ekstraksi rekam jejak label transisi sebelumnya (y_{prev}). Operasi $\text{grad_node} -= \text{val}$ dan $\text{grad_trans} -= 1.0$ digunakan untuk memasukkan nilai fitur dari data latih ke dalam matriks *gradien*. Sesuai dengan Persamaan (2.6) dan (2.7), *gradien* dihitung dengan mencari selisih antara observasi dan ekspektasi. Sistem dapat menghitung selisih matematis tersebut secara langsung ketika nilai prediksi mesin ditambahkan pada tahapan berikutnya.

```
for t in range(N):
    # Menghitung nilai gradien ke-k untuk fungsi fitur node (Gkx).
    for feat, val in x_seq[t].items():
        if feat in self.feature_to_idx:
            feat_id = self.feature_to_idx[feat]
            grad_node[feat_id, :] += node_marginals[t, :] * val

for t in range(N - 1):
    # Menghitung nilai gradien ke-k' untuk fungsi fitur edge (Gk'x).
    grad_trans += edge_marginals[t]
```

Gambar 4.25 Perhitungan Gradien Berdasarkan Probabilitas Marginal



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Operasi ini mengimplementasikan Persamaan (2.6) dan (2.7) untuk menghitung gradien sebagai selisih antara observasi dan ekspektasi label. Pada `grad_node`, nilai `node_marginals` dikalikan dengan `val` karena fitur `node` bergantung pada data masukan x_t , sedangkan pada `grad_trans`, `edge_marginals` ditambahkan langsung karena fitur `edge` hanya bergantung pada transisi label. Penambahan ekspektasi label ke dalam matriks *gradien* yang sebelumnya telah diinisialisasi dengan nilai observasi ini menghasilkan selisih antara observasi dan ekspektasi label.

```
# Rumus Log-Likelihood: Skor Empiris - Normalisasi Z(x)
log_likelihood += seq_score - Z_x

sigma = self.l2_reg
# Penalti untuk mencegah overfitting sigma / 2 * sigma lambda^2
reg_loss = (sigma / 2.0) * np.sum(weights ** 2)

# Menerapkan penalti regularisasi L2 pada gradien: Gk + sigma lambda k
grad_node += sigma * W_node
grad_trans += sigma * W_trans

# Mengubah Log-Likelihood menjadi Negative Log-Likelihood (karena L-BFGS-B mencari nilai minimum)
total_loss = -log_likelihood + reg_loss
total_grad = np.concatenate([grad_node.flatten(), grad_trans.flatten()])

return total_loss, total_grad
```

Gambar 4.26 Implementasi *Negative Log-Likelihood* dan Regularisasi L2

Nilai *log-likelihood* dihitung dengan mengurangi `seq_score` dengan fungsi normalisasi (`Z_x`). Penalti *L2 Regularization* dihitung melalui jumlah kuadrat bobot (`np.sum(weights ** 2)`) untuk memenuhi komponen fungsi penalti pada Persamaan (2.5) guna mencegah *overfitting*. Hasil tersebut dikonversi menjadi *negative log-likelihood* (`-log_likelihood + reg_loss`) agar sesuai dengan kebutuhan algoritma L-BFGS-B untuk mencari *loss* terendah.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
def _viterbi_decoding(self, node_potential, edge_potential):
    N = node_potential.shape[0]
    Y = self.num_labels

    # V menyimpan viterbi forward pass (alpha t^max)
    V = np.zeros((N, Y))
    # Menyimpan jejak index buat backtracking
    ptr = np.zeros((N, Y), dtype=int)

    log_node = np.log(np.clip(node_potential, 1e-100, None))
    log_edge = np.log(np.clip(edge_potential, 1e-100, None))

    # Inisialisasi nilai Viterbi forward pass pada urutan pertama (t=1)
    # Rumus:  $\alpha_1^{\max} [y_1] = \log(\phi_1(y_1))$ 
    V[0] = log_node[0]

    for t in range(1, N):
        # Menghitung nilai Viterbi forward pass maksimum pada urutan t >= 2 hingga T
        # Rumus:  $\alpha_t^{\max} [y_t] = \max_{y_{t-1}} (\alpha_{t-1}^{\max} [y_{t-1}] + \log \psi + \log \phi_t)$ 
        for y in range(Y):
            seq_probs = V[t-1] + log_edge[:, y] + log_node[t, y]
            V[t, y] = np.max(seq_probs) # Mencari probabilitas terbesar
            ptr[t, y] = np.argmax(seq_probs) # Menyimpan jalur asal dari probabilitas terbesar

    y_t_star = np.zeros(N, dtype=int)

    # Melakukan Viterbi backtracking untuk menentukan label akhir yang paling optimal (yt^*)
    # Rumus:  $y_t^* = \operatorname{argmax}_y (\alpha_t^{\max} [y_t])$ 
    y_t_star[N-1] = np.argmax(V[N-1])
    for t in range(N-2, -1, -1):
        y_t_star[t] = ptr[t+1, y_t_star[t+1]]

    return y_t_star
```

Gambar 4.27 Perhitungan *Viterbi Decoding*

Metode *Viterbi Decoding* mencari urutan label optimal melalui perhitungan *log-space* untuk menjaga stabilitas numerik. Proses dimulai dengan inisialisasi $V[0]$ sebagai titik pijakan awal sesuai Persamaan (2.14), kemudian dilanjutkan dengan *forward pass* pada Persamaan (2.15) yang menggunakan fungsi `np.max` untuk memilih probabilitas tertinggi, menyimpan skor ke dalam matriks V dan indeks jalur ke dalam ptr . Sesuai Persamaan (2.16), dilakukan *backtracking* untuk menyusun label optimal y_t_star dengan menelusuri matriks ptr secara mundur dari urutan akhir ke awal.

Konfigurasi parameter yang digunakan pada tahap pelatihan ini dijabarkan sebagai berikut:

Tabel 4. Parameter Pelatihan Model CRF

Nama Parameter	Nilai / Keterangan
----------------	--------------------



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Algorithm	L-BFGS
Regularization L2 (c2)	0.1
Max Iterations	200

Parameter algoritma ditetapkan pada L-BFGS-B (*Limited-memory Broyden–Fletcher–Goldfarb–Shanno with Box constraints*) sebagai metode optimasi. Untuk mengendalikan kompleksitas model, diterapkan koefisien regularisasi L2 (c2) yang bernilai 0.1. Penalti L2 ini berfungsi untuk mendistribusikan bobot secara merata dan mencegah model memberikan bobot terlalu dominan pada fitur-fitur tertentu, sehingga meminimalisir risiko model menghafal data pelatihan secara berlebihan (*overfitting*). Selain itu, batasan `max_iterations` diatur ke angka 200 untuk memastikan algoritma L-BFGS-B memiliki kesempatan komputasi yang cukup untuk menyusuri gradien dan mencari titik *loss* terendah sebelum proses iterasi dihentikan secara sistem.

```

MIN_FUZZY_TOKEN_LENGTH = 4
MAX_EDIT_DISTANCE = 2

def edit_distance(left: str, right: str) -> int:
    if left == right:
        return 0
    if not left:
        return len(right)
    if not right:
        return len(left)
    previous_row = list(range(len(right) + 1))
    for left_index, left_char in enumerate(left, start=1):
        current_row = [left_index]
        for right_index, right_char in enumerate(right, start=1):
            insert_cost = current_row[right_index - 1] + 1
            delete_cost = previous_row[right_index] + 1
            replace_cost = previous_row[right_index - 1] + (left_char != right_char)
            current_row.append(min(insert_cost, delete_cost, replace_cost))
        previous_row = current_row
    return previous_row[-1]

def get_common_prefix_length(left: str, right: str) -> int:
    shared = 0
    for left_char, right_char in zip(left, right):
        if left_char != right_char:
            break
        shared += 1
    return shared

def find_best_token_correction(token: str, vocabulary: set[str]) -> dict[str, object] | None:
    normalized_token = normalize_token(token)
    if len(normalized_token) < MIN_FUZZY_TOKEN_LENGTH:
        return None

```

Gambar 4.28 Fuzzy String Matching



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Meskipun model CRF mampu memprediksi letak entitas lokasi dengan presisi tinggi, prediksi yang dihasilkan seringkali masih berupa kata mentah yang mengandung kesalahan ejaan (*typo*) dari pengguna. Karena algoritma prediksi tidak boleh dibiarkan memberikan keluaran yang salah ke sistem pencarian *database*, maka dibangunlah sistem *entity resolution* sebagai modul penyaring akhir di dalam arsitektur model secara keseluruhan.

Sistem resolusi ini mengandalkan algoritma *Fuzzy String Matching* berbasis kalkulasi *Levenshtein Distance*. Integrasi algoritma *Levenshtein Distance* sangat efektif dan memiliki reliabilitas tinggi (mencapai presisi 92%) dalam mendeteksi serta mengoreksi variasi kesalahan ejaan pada struktur bahasa dokumen yang kompleks (Santoso dan Yan, 2024). Algoritma ini bekerja dengan menghitung jumlah operasi karakter (penambahan, penghapusan, atau penggantian huruf) yang dibutuhkan untuk mengonversi *typo* pengguna menjadi teks resmi di dalam *gazetteer*. Fungsi `find_best_token_correction` bertugas mengevaluasi kata anomali yang diprediksi oleh CRF, mengukurnya dengan seluruh kosakata *knowledge base*, dan secara dinamis menarik kandidat terdekat selama jumlah kesalahannya tidak melebihi batas toleransi ($\text{Max Edit Distance} = 2$). Mekanisme ini krusial untuk memastikan chatbot tetap responsif dan *fault-tolerant* di tengah variasi penulisan informal komunitas.

Dilakukan *white-box testing* untuk memvalidasi proses matematis internal model secara manual, mulai dari pembentukan vektor fitur hingga tahap inferensi. Penjabaran langkah demi langkah untuk mendemonstrasikan pemahaman mendalam mengenai kalkulasi matriks algoritma CRF akan dijelaskan dibagian selanjutnya.

WHITE-BOX TESTING - LinearChainCRF

```
[1] INISIALISASI DATA
Kalimat masukan (x) : ['how', 'to', 'enter', 'liyue', 'city']
Panjang kalimat (T) : 5
Target kata yang dibedah (t) : 4
Target label sampel dibedah : 'B-LOC'
```

Gambar 4.29 Inisialisasi Data untuk *White-Box Testing*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Simulasi komputasi ini berfokus pada kata ke-4 ($t = 4$), yaitu 'liyue', dari observasi masukan $x = ['how', 'to', 'enter', 'liyue', 'city']$ (dengan total panjang $T = 5$).

Target pelacakan label pada posisi ini adalah $y_4 = B - LOC$.

```
[2] POTENSIAL NODE (phi t(yt, x)) PADA t=4 (kata: 'liyue')
Daftar fitur aktif dan nilai bobot lambda k-nya:
- bias : [0= 2.237, B-LOC=-1.384, I-LOC=-0.853]
- word.prefix2=li : [0=-0.554, B-LOC= 0.686, I-LOC=-0.132]
- word.prefix3=liy : [0=-0.232, B-LOC= 0.254, I-LOC=-0.022]
- word.suffix2=ue : [0= 0.411, B-LOC=-0.116, I-LOC=-0.295]
- word.suffix3=yue : [0=-0.232, B-LOC= 0.254, I-LOC=-0.022]
- postag=JJ : [0=-0.383, B-LOC=-0.081, I-LOC= 0.464]
- postag[:2]=JJ : [0=-0.467, B-LOC= 0.134, I-LOC= 0.333]
- is_in_genshin_map=True : [0=-1.295, B-LOC= 0.592, I-LOC= 0.703]
- gazetteer_word_overlap=True : [0=-1.295, B-LOC= 0.592, I-LOC= 0.703]
- gazetteer_phrase_overlap=True : [0=-2.060, B-LOC= 1.290, I-LOC= 0.770]
- word.trigram:liy=True : [0=-0.232, B-LOC= 0.254, I-LOC=-0.022]
- word.trigram:iyu=True : [0=-0.232, B-LOC= 0.254, I-LOC=-0.022]
- word.trigram:yue=True : [0=-0.232, B-LOC= 0.254, I-LOC=-0.022]
- -1:postag=VB : [0= 0.364, B-LOC=-0.081, I-LOC=-0.283]
- -1:postag[:2]=VB : [0= 0.079, B-LOC=-0.027, I-LOC=-0.051]
- +1:postag=NN : [0= 0.136, B-LOC= 0.300, I-LOC=-0.437]
- +1:postag[:2]=NN : [0= 0.115, B-LOC=-0.068, I-LOC=-0.048]
- +1:gazetteer_word_overlap=True : [0=-0.483, B-LOC= 0.658, I-LOC=-0.176]
```

Bedah Rumus Potensial Node untuk label 'B-LOC' pada t=4:
 sum (lambda k * f k) untuk 'B-LOC' = 3.7667

Gambar 4.30 Hasil Perhitungan Fitur Aktif Dengan Bobot

Sebelum probabilitas dihitung, observasi x pada posisi $t = 4$ dipetakan ke dalam bentuk boolean oleh fungsi fitur state $f_k(y_t, x, t)$ dan fungsi fitur transisi $f_{k'}(y_t, y_{t+1}, x, t)$. Setiap fitur yang bernilai 1 (aktif) kemudian dikalikan dengan parameter bobotnya masing-masing (λ_k dan $\lambda_{k'}$). Untuk target kelas label $y_4 = B - LOC$, akumulasi perkalian matriks bobot dan fitur ($\sum_k \lambda_k f_k$) menghasilkan skor skalar sebesar 3.7667.

```
Shift (pengurang max untuk hindari overflow) = 3.7667
phi 4['B-LOC'] = exp(3.7667 - 3.7667) = 1.0000
Matriks lengkap phi 4:
- phi 4['O'] = 0.0003
- phi 4['B-LOC'] = 1.0000
- phi 4['I-LOC'] = 0.0418
```

Gambar 4.31 Perhitungan Node Potential



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Nilai probabilitas asosiasi antara kata 'liyue' dengan label B-LOC dieksekusi menggunakan Persamaan (2.3). Untuk mencegah *numerical overflow* pada limitasi memori komputer, diterapkan pergeseran nilai (*shift*) sebesar nilai maksimumnya (3.7667): $\phi_4(B - LOC, x) = \exp(3.7667 - 3.7667) = 1.0000$ Distribusi ϕ_4 untuk seluruh himpunan y menghasilkan: ['O': 0.0003, 'B-LOC': 1.0000, 'I-LOC': 0.0418].

```
[3] POTENSIAL EDGE (psi(y, yt+1, x))
Matriks bobot lambda k' (Baris = Dari, Kolom = Ke):
      0      B-LOC      I-LOC
0      4.1148      3.0778      -5.5894
B-LOC  -1.9089     -1.9168      2.7793
I-LOC  -0.8554     -2.1468      2.4455

Bedah Rumus Potensial Edge untuk sampel transisi 'B-LOC' -> 'I-LOC':
lambda k' ('B-LOC' -> 'I-LOC') = 2.7793
Shift = 4.1148
psi('B-LOC' -> 'I-LOC') = exp(2.7793 - 4.1148) = 0.2630
Matriks lengkap psi:
- Dari 'O': [ 1.0000  0.3545  0.0001]
- Dari 'B-LOC': [ 0.0024  0.0024  0.2630]
- Dari 'I-LOC': [ 0.0069  0.0019  0.1884]
```

Gambar 4.32 Perhitungan *Potential Edge*

Perpindahan dari satu label ke label berikutnya diukur menggunakan Persamaan (2.4). Jika kita mengambil sampel probabilitas transisi dari kelas B-LOC menuju 'I-LOC' ($\sum \lambda_{k'} = 2.7793$ dan *shift* maksimum 4.1148): $\psi_t(B - LOC, I - LOC, x) = \exp(2.7793 - 4.1148) = 0.2630$.

```
[4] FORWARD PASS (alpha t)
Bedah Rumus Forward Pass untuk alpha 4['B-LOC'] (t=4):
alpha 4['B-LOC'] = phi 4['B-LOC'] * [ (alpha_3['O'] * psi('O' -> 'B-LOC')) + (alpha_3['B-LOC'] * psi('B-LOC' -> 'B-LOC')) + (alpha_3['I-LOC'] * psi('I-LOC' -> 'B-LOC')) ]
alpha 4['B-LOC'] = 1.0000 * [ (0.9992 * 0.3545) + (0.0008 * 0.0024) + (0.0000 * 0.0019) ]
alpha 4['B-LOC'] = 1.0000 * 0.3542 = 0.3542
(Catatan: Nilai aktual di dalam memori model di-scaling/dibagi untuk mencegah underflow. Nilai akhirnya = 0.9991)
```

Gambar 4.33 Perhitungan Forward Pass

Untuk menghitung peluang sekuensial dari $t = 1$ hingga mencapai $t = 4$, model menerapkan Persamaan (2.9). Kalkulasi ini mengalikan probabilitas *node potential* saat ini dengan akumulasi *forward pass* urutan sebelumnya yang telah dikalikan *edge potential* untuk mendapatkan nilai mentah:

$$\begin{aligned} \alpha_4[B - LOC] = & \phi_4(B - LOC) [(\alpha_3[O] \psi(O \rightarrow B - LOC)) \\ & + (\alpha_3[B - LOC] \psi(B - LOC \rightarrow B - LOC)) \\ & + (\alpha_3[I - LOC] \psi(I - LOC \rightarrow B - LOC))] \end{aligned}$$



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

$$\alpha_4[B - LOC] = 1.0000 [(0.9992 * 0.3545) + (0.0008 * 0.3545) + (0.0000 * 0.0019)]$$

$$\alpha_4[B - LOC] = 1.0000 [(0.9992 * 0.3545) + (0.0008 * 0.3545) + (0.0000 * 0.0019)]$$

$$\alpha_4[B - LOC] = 1.0000 * 0.3542 = 0.3542$$

Untuk mencegah *numerical underflow* di dalam memori komputer, nilai mentah tersebut dinormalisasi dengan membaginya terhadap konstanta skala pembagi (k_4). Nilai k_4 didapatkan dari total penjumlahan probabilitas mentah seluruh kelas pada posisi observasi ke-4 ($\alpha_4[O] + \alpha_4[B - LOC] + \alpha_4[I - LOC] = 0.3545$) sehingga probabilitas *scaled* akhirnya bernilai:

$$\alpha_4[B - LOC] = \frac{0.3542}{0.3545} = 0.9991$$

```
[5] BACKWARD PASS (beta t)
Berdasarkan Rumus Backward Pass untuk beta 4["B-LOC"] (t=4):
beta 4 ["B-LOC"] = (beta 5["O"] * phi 5["O"] * psi("B-LOC" -> "O")) + (beta 5["B-LOC"] * phi 5["B-LOC"] * psi("B-LOC" -> "B-LOC")) + (beta 5["I-LOC"] * phi 5["I-LOC"] * psi("B-LOC" -> "I-LOC"))
beta 4 ["B-LOC"] = (0.3333 * 0.0005 * 0.0024) + (0.3333 * 0.0192 * 0.0024) + (0.3333 * 1.0000 * 0.2630)
beta 4 ["B-LOC"] = 0.0877
(Catatan: Nilai aktual memori model setelah scaling/dibagi mu t = 0.5733)
```

Gambar 4.34 Perhitungan Backward Pass

Karena posisi kata $t = 4$ lebih kecil dari total panjang kalimat $T = 5$, penelusuran mundur tidak menggunakan Persamaan (2.10), melainkan Persamaan (2.11). Probabilitas ditelusuri secara mundur dari kata 'city' ($t = 5$) menuju 'liyue' ($t = 4$). Kalkulasi ini menjumlahkan hasil kali antara probabilitas transisi (*edge potential*), probabilitas langkah berikutnya (*node potential* pada $t + 1$), dan akumulasi *backward pass* langkah berikutnya (β_4) untuk mendapatkan nilai mentah *beta*.

$$\begin{aligned} \beta_4[B - LOC] = & (\beta_5[O] \phi_5(O) \psi(B - LOC \rightarrow O)) \\ & + (\beta_5[B - LOC] \phi_5(B - LOC) \psi(B - LOC \rightarrow B - LOC)) \\ & + (\beta_5[I - LOC] \phi_5(I - LOC) \psi(B - LOC \rightarrow I - LOC)) \end{aligned}$$

$$\beta_4[B - LOC] = (0.3333 * 0.0005 * 0.0024) + (0.3333 * 0.0192 * 0.0024) + (0.3333 * 1.0000 * 0.2630)$$

$$\beta_4[B - LOC] = 0.0877$$

Sama halnya dengan *Forward Pass*, untuk mencegah *numerical underflow* di dalam memori komputer, nilai mentah tersebut dinormalisasi dengan membaginya terhadap konstanta skala pembagi (μ_4). Nilai μ_4 didapatkan dari total penjumlahan probabilitas mentah penelusuran mundur seluruh kelas pada posisi observasi ke-4



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

$(\beta_4[O] + \beta_4[B - LOC] + \beta_4[I - LOC] = 0.1530)$, sehingga probabilitas *scaled* akhirnya bernilai:

$$\beta_4[B - LOC] = \frac{0.0877}{0.1530} = 0.5733$$

```
[6] PROBABILITAS MARGINAL (P(yt | x))
Nilai normalisasi Z(x) (dalam log-space) = 36.0472

Bedah Rumus Probabilitas Marginal Node untuk label 'B-LOC' pada t=4:
P(y4='B-LOC' | x) = (alpha 4['B-LOC'] * phi 4['B-LOC'] * beta 4['B-LOC']) / Normalisasi Z
P(y4='B-LOC' | x) = (0.9991 * 1.0000 * 0.5733) / Z(x)
Hasil probabilitas akhir model = 1.0000 (atau 100.00%)
Probabilitas Marginal lengkap t=4:
- P(y4='O' | x) = 0.0000
- P(y4='B-LOC' | x) = 1.0000
- P(y4='I-LOC' | x) = 0.0000
```

Gambar 4.35 Perhitungan Probabilitas Marginal

Setelah penelusuran dua arah selesai, peluang riil bahwa 'liyue' adalah representasi dari entitas B-LOC ditentukan melalui Persamaan (2.12). Hasil kali α_4 dan β_4 kemudian dibagi dengan konstanta partisi normalisasi $Z(x)$ yang didapat dari Persamaan (2.2).

Nilai $Z(x)$ dalam *log-space* merupakan total jumlahan dari seluruh faktor skala pada proses *Forward Pass*, yaitu $\log Z(x) = \sum_{t=1}^T \log k_t$ yang bernilai 36.0472. Namun pada implementasi yang nilai α dan β -nya telah dinormalisasi (*scaled*), konstanta $Z(x)$ lokal pada posisi t dapat dihitung secara efisien dengan menjumlahkan seluruh kemungkinan hasil kali α dan β untuk semua kelas label Y (yaitu O, B-LOC, I-LOC):

$$Z(x)_t = \sum_{y \in Y} \alpha_t(y) \phi_t(y) \beta_t(y)$$

$$Z(x)_4 = (\alpha_4[O] * 0.0003 * \beta_4[O]) + (0.9991 * 1.0000 * 0.5733) + (\alpha_4[I - LOC] * 0.0418 * \beta_4[I - LOC])$$

Berdasarkan *output* probabilitas marginal akhir di mana nilai observasi untuk label O dan 'I-LOC' jatuh pada angka 0.0000, dapat dibuktikan secara matematis bahwa matriks nilai α dan β untuk kedua label tersebut tereduksi mendekati nol. Oleh



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

karena itu, akumulasi konstanta pembagi $Z(x)_4$ secara dominan murni oleh perhitungan label B-LOC: $Z(x)_4 \approx (0.9991 * 1.0000 * 0.5733) = 0.5727$

Probabilitas marginal untuk 'B-LOC' dikalkulasi dari rasio tersebut:

$$P_4(B-LOC|x) \frac{\alpha_t(B-LOC)\phi_t(B-LOC)\beta_t(B-LOC)}{Z(x)_4}$$

$$P_4(B-LOC|x) \frac{0.5727}{0.5727} = 1.0000$$

Hasil evaluasi ini menempatkan probabilitas akhir P_4 untuk B-LOC secara absolut pada angka 1.0000 (100.00%).

```
[7] VITERBI DECODING (Jalur Optimal y^*)
Bedah Rumus Viterbi pada t=4 (kata: 'liyue') untuk label 'B-LOC':
Kalkulasi probabilitas kumulatif (log-space) dari setiap kemungkinan jalur sebelumnya (t-1):
- Dari jalur 'O': V[3] (0.0000) + log psi (-1.0370) + log phi (0.0000) = -1.0370
- Dari jalur 'B-LOC': V[3] (-7.0805) + log psi (-6.0317) + log phi (0.0000) = -13.1122
- Dari jalur 'I-LOC': V[3] (-14.6076) + log psi (-6.2617) + log phi (0.0000) = -20.8693
Nilai maksimum terpilih = -1.0370 (berasal dari jalur 'O')
Maka, V[4]['B-LOC'] = -1.0370, dan PTR (Backpointer) menunjuk ke 'O'

Proses Penentuan Argmax pada t=5 (Posisi Akhir Kalimat):
- Skor Kumulatif Viterbi untuk 'O' = -14.6154
- Skor Kumulatif Viterbi untuk 'B-LOC' = -11.0211
- Skor Kumulatif Viterbi untuk 'I-LOC' = -2.3725
Nilai Maksimum terpilih jatuh pada label: 'I-LOC'

Melalui backtracking pada PTR (Pointer Viterbi), jalur label kelas optimal y^* untuk kalimat 'how to enter liyue city' adalah:
--> ['O', 'O', 'B-LOC', 'I-LOC']
```

Gambar 4.36 Perhitungan Viterbi Decoding

Tahap akhir diakhiri dengan Viterbi *decoding* untuk menemukan urutan label y yang paling maksimal secara global untuk observasi x . Proses ini dimulai dengan *Maximal Forward Pass* (Persamaan 2.15), di mana model mengevaluasi probabilitas kumulatif maksimum (dalam *log-space*) dari setiap kemungkinan jalur sebelumnya ($t - 1$) menuju posisi saat ini. Model menjumlahkan skor kumulatif sebelumnya ($V[3]$) dengan probabilitas *edge* ($\log \psi$) dan probabilitas *node* ($\log \phi$):

$$V[t-1] + \log(\phi_t(y_t, x)) + \log \psi_t((y_{t-1}, y_t, x))$$

Dari jalur O:

$$0.0000 + (-1.0370) + 0.0000 = -1.0370$$

Dari jalur B-LOC:

$$-7.0805 + (-6.0317) + 0.0000 = -13.1122$$

Dari jalur I-LOC:



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

$$-14.6076 + (-6.2617) + 0.0000 = -20.8693$$

Nilai maksimum terpilih adalah -1.0370 yang berasal dari transisi jalur O. Maka, nilai $V[4][B - LOC] = -1.0370$, dan algoritma menyimpan *backpointer* yang menunjuk ke label O.

Tahap akhir dari *Viterbi Decoding* adalah menentukan pemenang global di posisi akhir kalimat ($t = 5$) dan menarik mundur jejaknya. Berdasarkan hasil iterasi *Maximal Forward Pass*, skor kumulatif Viterbi pada posisi $t = 5$ adalah sebagai berikut:

- Skor Kumulatif Viterbi untuk O = -14.6154
- Skor Kumulatif Viterbi untuk B-LOC = -11.0211
- Skor Kumulatif Viterbi untuk I-LOC = -2.3725

Nilai maksimum mutlak jatuh pada label I-LOC. Label inilah yang menjadi titik awal untuk proses *Backtracking Pass* menggunakan Persamaan (2.16). Dengan menerapkan fungsi $\arg \max_{y_t}$ untuk menelusuri mundur rekam jejak *pointer* dari garis akhir tersebut, sistem menarik jejak mundur sebagai berikut:

$$I - LOC(t = 5) \rightarrow B - LOC(t = 4) \rightarrow O(t = 3) \rightarrow O(t = 2) \rightarrow O(t = 1)$$

Berdasarkan keseluruhan rantai persamaan di atas, perumusan matematis berhasil menarik kesimpulan final bahwa y^* yang paling optimal untuk kalimat tersebut adalah ['O', 'O', 'O', 'B-LOC', 'I-LOC'].

4.3.5 Evaluation

Tabel 5. Hasil Pengujian Model CRF dengan Augmentasi Data

Test Size	70:30	80:20	90:10
B-LOC			
Precision	0.9872	0.9905	0.9811
Recall	0.9747	0.9905	0.9811
F1-Score	0.9809	0.9905	0.9811
I-LOC			
Precision	0.9897	0.9925	0.9844
Recall	1.0000	1.0000	1.0000
F1-Score	0.9948	0.9962	0.9921



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Weighted Avg			
Precision	0.9886	0.9916	0.9829
Recall	0.9886	0.9958	0.9914
F1-Score	0.9885	0.9937	0.9871
Accuracy	0.9969	0.9984	0.9970

Tabel 6. Hasil Pengujian Model CRF Tanpa Augmentasi Data

Test Size	70:30	80:20	90:10
B-LOC			
Precision	0.3333	0.5000	1.0000
Recall	0.1429	0.2000	0.6667
F1-Score	0.2000	0.2857	0.8000
I-LOC			
Precision	0.6667	0.5000	1.0000
Recall	0.1250	0.1000	0.1667
F1-Score	0.2105	0.1667	0.2857
Weighted Avg			
Precision	0.5652	0.5000	1.0000
Recall	0.1304	0.1333	0.3333
F1-Score	0.2073	0.2063	0.4571
Accuracy	0.7375	0.7500	0.8462

Tahap evaluasi pemodelan dilakukan dengan membandingkan metrik kinerja algoritma *Conditional Random Field* (CRF) dalam mengekstrak entitas nama lokasi (*Location*) menggunakan *dataset* asli dan *dataset* yang telah melalui proses augmentasi. Hasil pengujian klasifikasi terhadap *tag* B-LOC dan I-LOC membuktikan bahwa penerapan teknik augmentasi data memberikan peningkatan performa yang sangat signifikan. Model tanpa augmentasi menunjukkan kinerja yang fluktuatif dan kurang optimal, ditandai dengan nilai *recall* keseluruhan yang sangat rendah di angka 0.1304 pada rasio pembagian data 70:30. Hal ini mengindikasikan bahwa jumlah variasi data latih mentah yang terlalu sedikit membuat model kesulitan dan kehilangan kemampuan mendeteksi entitas saat dihadapkan pada kalimat uji yang tidak pernah dilihat sebelumnya.

Pengujian model dengan augmentasi data menunjukkan tingkat stabilitas pengenalan entitas yang sangat baik secara konsisten di seluruh skenario pembagian data (70:30, 80:20, maupun 90:10). Skenario pembagian data latih dan data uji sebesar 80:20 dengan augmentasi diidentifikasi sebagai model dengan tingkat



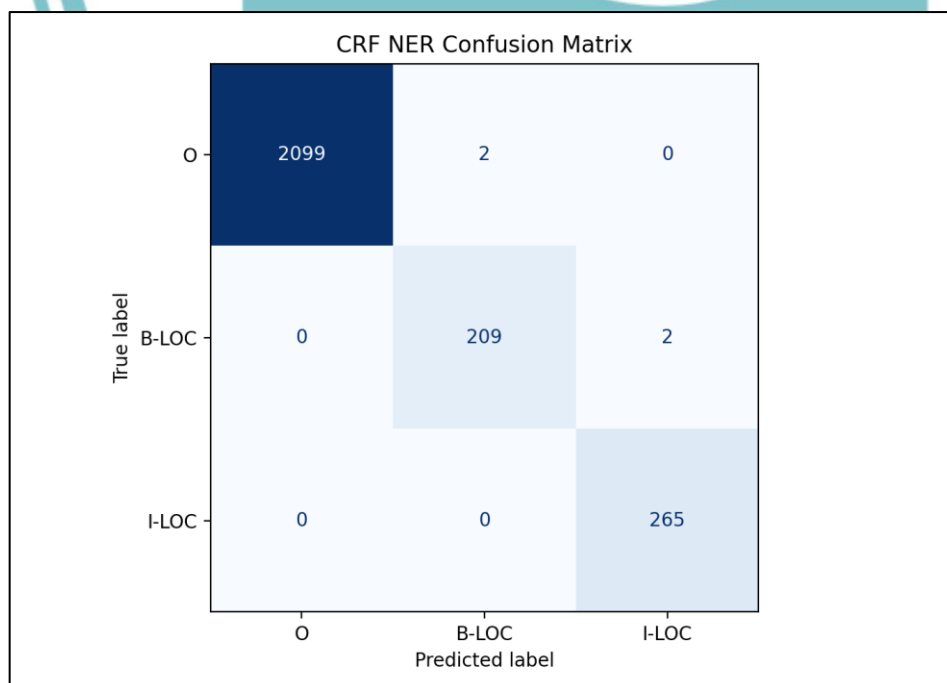
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

performa paling superior dan tervalidasi dengan baik. Pada rasio pembagian tersebut, model berhasil mencetak tingkat *accuracy* tertinggi mencapai angka 0.9984 (99,84%).

Kinerja klasifikasi model yang sangat akurat ini dibuktikan oleh visualisasi *Confusion Matrix* pada Gambar 4.37. Matriks tersebut memetakan perbandingan secara rinci antara label prediksi dari sistem (*predicted label*) dengan label aktual (*true label*) untuk tag O (bukan lokasi), B-LOC (awal nama lokasi), dan I-LOC (lanjutan nama lokasi). Data pada matriks menunjukkan bahwa model berhasil memprediksi kelas secara tepat (*True Positive*) dengan jumlah yang sangat dominan, yaitu 2.099 *token* untuk kelas O, 209 *token* untuk B-LOC, dan 265 *token* untuk I-LOC. Tingkat misklasifikasi terpantau sangat minim, di mana kesalahan prediksi (*False Positive* maupun *False Negative*) hanya terjadi pada segelintir *token*, seperti 2 *token* kelas O yang terprediksi sebagai B-LOC, dan 2 *token* kelas B-LOC yang terprediksi sebagai I-LOC. Minimnya angka kesalahan tebakan ini secara visual mempertegas tingginya kepastian model dalam membedakan batas-batas entitas lokasi di dalam struktur kalimat yang bervariasi.



Gambar 4.37 *Confusion Matrix* Test Size 80:20

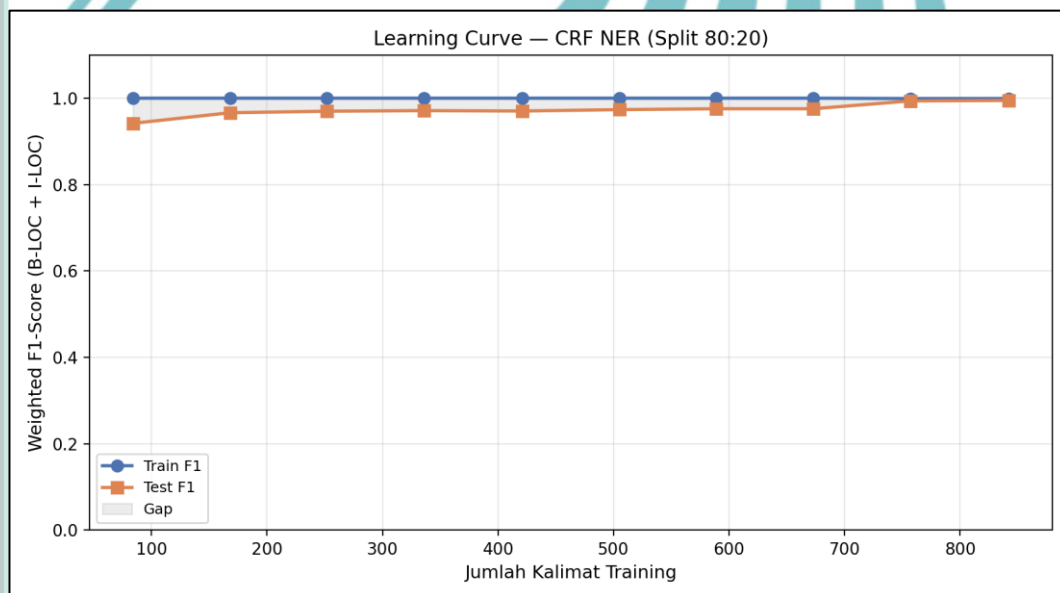


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Keseimbangan kemampuan model pada skenario 80:20 tersebut semakin dipertegas oleh perolehan metrik evaluasi lainnya yang dihitung berdasarkan *weighted average*. Model mampu memprediksi entitas dengan *precision* sebesar 0.9916 (99,16%) dan *recall* sebesar 0.9958 (99,58%), yang secara seimbang pada perolehan *F1-Score* sebesar 0.9937 (99,37%). Pencapaian ini membuktikan bahwa strategi augmentasi pada tahapan *data preparation* sangat krusial dalam mengatasi kekurangan data mentah dan menyimulasikan kerumitan bahasa tidak terstruktur pemain *Genshin Impact*. Tingginya nilai evaluasi ini memastikan bahwa arsitektur model CRF yang dibangun sangat andal untuk diintegrasikan ke dalam layanan *chatbot*, sehingga sistem dapat mengidentifikasi entitas lokasi secara presisi sebagai kunci utama untuk mencari daftar *quest* prasyarat.



Gambar 4.38 Learning Curve CRF dengan *Test Size* 80:20

Analisis terhadap pergerakan kemampuan generalisasi model CRF pada skenario 80:20 divisualisasikan melalui grafik *Learning Curve* pada Gambar 4.38. Grafik ini memetakan nilai *Weighted F1-Score* terhadap peningkatan jumlah kalimat *training* untuk mendiagnosis indikasi *overfitting* maupun *underfitting* pada model. Kurva pengujian (*Test F1*) mengawali performa di angka 0.9418 sejak fase awal pelatihan dengan masukan 84 kalimat (10% dari total data *training*). Seiring dengan bertambahnya volume data, kurva pengujian tersebut secara umum menunjukkan tren yang cenderung naik, meskipun disertai dengan sedikit fluktuasi minor.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Sementara itu kurva pelatihan (*Train F1*) mayoritas bertahan mutlak di angka 1.0000 sebelum akhirnya sedikit menyesuaikan di angka 0.9992 pada iterasi akhir.

Tingginya skor evaluasi pengujian yang langsung berada di atas angka 0.94 sejak awal fase pelatihan ($n=84$) secara otomatis menyingkirkan kemungkinan terjadinya *underfitting* atau kegagalan model dalam belajar. Namun rentang celah (*gap*) antara *Train F1* dan *Test F1* pada fase penggunaan 10% hingga 80% data ($n=84$ hingga 673) yang membelah jarak sekitar 0.0245 hingga 0.0582 mengindikasikan adanya fenomena *overfitting* ringan. Pada fase ini, model mampu menghafal data latih secara absolut (skor 1.0000), sementara performa pada data uji sedikit tertinggal.

Titik paling krusial dari analisis kurva ini justru terletak pada lompatan peningkatan generalisasi yang sangat signifikan pada masa transisi antara 80% ke 90% data latih, di mana *gap* menyusut drastis dari 0.0245 menjadi 0.0055. Pada titik akhir penggunaan 100% data (842 kalimat), celah tersebut semakin menipis dan berakhir di angka 0.0045. Konvergensi *gap* yang meruncing tajam di fase akhir ini menjadi bukti bahwa penambahan kuantitas data latih secara utuh berhasil menekan *overfitting* ringan secara efektif. Tingginya angka generalisasi model ini sangat didukung oleh karakteristik data prasyarat *quest Genshin Impact* yang cenderung homogen, terstruktur, dan berulang, sehingga memudahkan algoritma CRF untuk mengekstrak pola leksikal dari entitas target dengan probabilitas akurasi yang maksimal.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3.6 Deployment

Model AI yang sebelumnya berupa struktur data statis diintegrasikan ke dalam arsitektur perangkat lunak yang utuh agar dapat berinteraksi langsung dengan pengguna.

A. Flask sebagai Layanan API

```
BASE_DIR = Path('../').resolve()
GAZETTEER_PATH = BASE_DIR / 'genshin_areas.json'
MODEL_PATH = BASE_DIR / 'outputs' / 'genshin_location_crf.pkl'

print(f"Loading Gazetteer from {GAZETTEER_PATH}")
gazetteer_names, gazetteer_tokens = load_gazetteer(str(GAZETTEER_PATH))

print(f"Loading Model from {MODEL_PATH}")
import __main__
__main__.LinearChainCRF = LinearChainCRF
with open(MODEL_PATH, 'rb') as f:
    crf_model = pickle.load(f)

@app.route('/extract', methods=['POST'])
def extract():
    data = request.get_json()
    if not data or 'text' not in data:
        return jsonify({'error': 'Missing text field'}), 400

    text = data['text']
    try:
        locations = extract_locations(text, crf_model, gazetteer_names, gazetteer_tokens)
        return jsonify({'locations': locations})
    except Exception as e:
        import traceback
        traceback.print_exc()
        return jsonify({'error': str(e)}), 500

if __name__ == '__main__':
    port = int(os.environ.get('PORT', 5001))
    app.run(host='0.0.0.0', port=port)
```

Gambar 4.39 Potongan Kode app.py

Pada Gambar 4.18, Flask membuka endpoint jaringan di POST /extract. Saat aplikasi utama mengirimkan teks, fungsi extract() akan mengeksekusi fungsi extract_locations() yang memuat seluruh pipeline AI (mulai dari CRF hingga *Fuzzy String Matching*). Hasilnya (berupa daftar entitas lokasi yang telah bersih) akan dibungkus ke dalam format JSON dan dikembalikan kepada client. Environment isolation menggunakan Flask ini menjamin stabilitas library NLP Python tanpa membebani server utama.



B. Integrasi dengan Framework Laravel

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
public function checkQuest(CheckQuestRequest $request)
{
    $userInput = $request->input('area_name');

    $extractedArea = null;
    try {
        $nerUrl = env('NER_SERVICE_URL', 'http://127.0.0.1:5001/extract');
        if (strpos($nerUrl, '/extract')) {
            $nerUrl = rtrim($nerUrl, '/') . '/extract';
        }
        $nerResponse = Http::timeout(30)->post($nerUrl, [
            'text' => $userInput
        ]);

        if ($nerResponse->successful() && isset($nerResponse->json()['locations'])) {
            $locations = $nerResponse->json()['locations'];
            if (!empty($locations)) {
                $extractedArea = $locations[0];
            }
        }
    } catch (\Exception $e) {
    }

    if (!$extractedArea) {
        return response()->json([
            'found' => false,
            'message' => "Hmm... Paimon couldn't detect any location in your message. Try asking about a location available in Genshin Impact, like \"Enkanomiya\"",
            'ner_used' => true,
        ], 404);
    }
}
```

Gambar 4.40 Mengirim *User Input* 'text' ke Flask

Saat pengguna menginput pertanyaan pada antarmuka, fungsi `Http::post` akan meneruskan teks tersebut secara internal (*backend-to-backend*) menuju layanan Flask. Laravel akan mengekstrak variabel hasil (`$locations[0]`) dari paket JSON yang direlaskan oleh Flask. Variabel inilah yang bertindak sebagai entitas hasil pengolahan kecerdasan buatan.

```
$regionAreas = AreaPrerequisite::whereRaw('LOWER(region) = ?', [strtolower($extractedArea)])
    ->where('location_type', 'Area')
    ->pluck('area_name');

if ($regionAreas->isEmpty()) {
    return response()->json([
        'found' => true,
        'is_region' => true,
        'region_name' => ucwords($extractedArea),
        'areas' => $regionAreas
    ]);
}

$query = AreaPrerequisite::query()
    ->byAreaName($extractedArea);

if ($request->filled('region')) {
    $query->byRegion($request->input('region'));
}

$result = $query->first();

if (!$result) {
    return response()->json([
        'found' => false,
        'message' => "Paimon couldn't find an area named \"{$extractedArea}\" in the knowledge base.",
    ], 404);
}

return new AreaPrerequisiteResource($result);
```

Gambar 4.41 Ekstraksi *Database* Untuk Quest Prasyarat



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

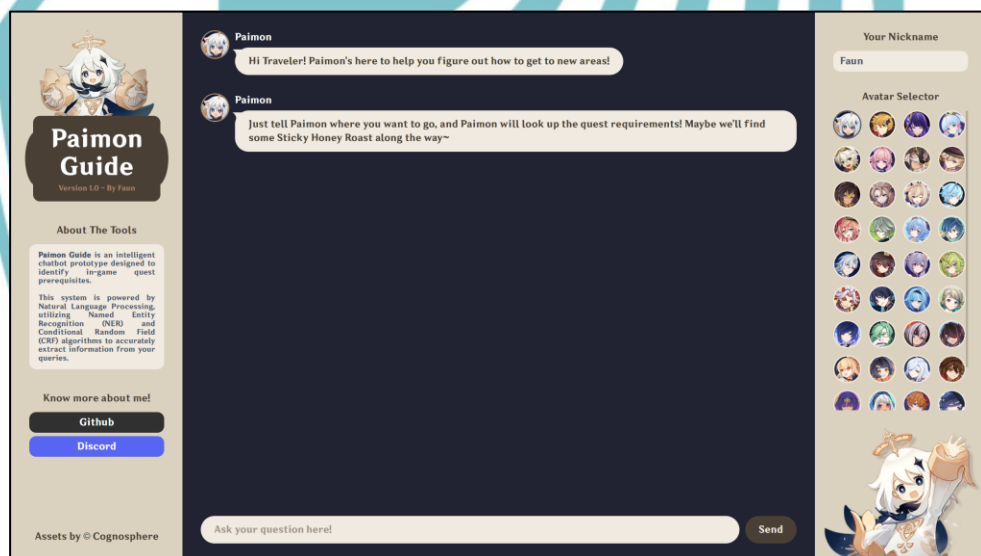
Laravel kemudian menggunakan variabel \$extractedArea tersebut sebagai *Query Parameter* untuk mengeksekusi pencarian pada tabel AreaPrerequisite di dalam *database* SQLite. Data detail yang didapatkan (seperti syarat quest wilayah) kemudian dikembalikan ke antarmuka pengguna dalam bentuk JSON Resource.

4.4 Implementasi Sistem

4.4.1 Implementasi Chatbot

Berikut implementasi interface dari website *chatbot* “Paimon Guide”:

A. Tampilan Awal



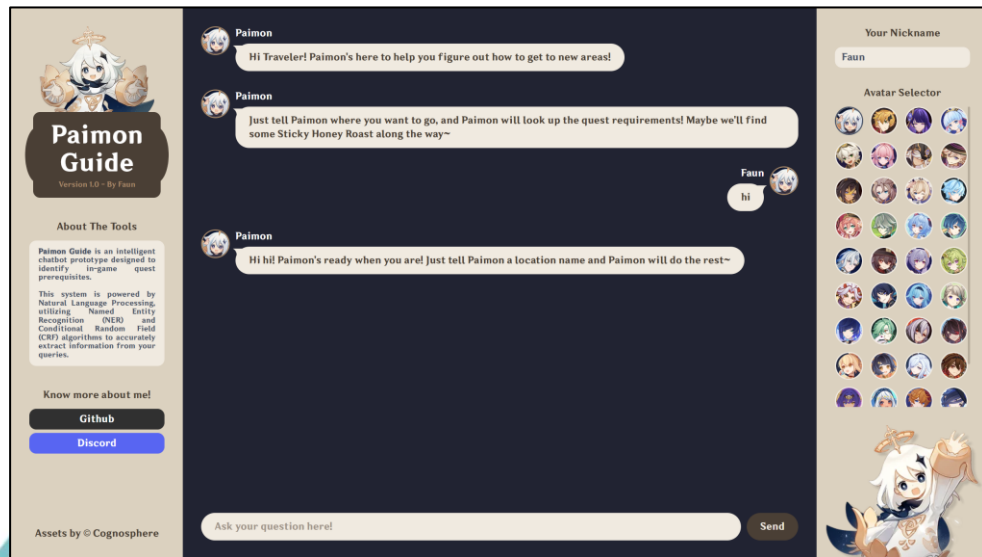
Gambar 4.42 Tampilan Awal Paimon Guide

Gambar 4.42 menunjukkan antarmuka beranda yang ditampilkan saat pengguna pertama kali mengakses aplikasi web Paimon Guide. Pada tahap ini, antarmuka memuat panel obrolan di bagian tengah yang diawali dengan pesan sapaan otomatis dari *chatbot*, serta informasi aplikasi dan fitur pengaturan profil di sisi kanan. Desain antarmuka ini memungkinkan pengguna untuk membaca deskripsi sistem atau langsung melakukan personalisasi profil sebelum memulai interaksi pencarian informasi quest.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

B. Menyapa Paimon



Gambar 4.43 Mengirim Sapaan "hi"

Gambar 4.43 mendemonstrasikan respons chatbot ketika menerima masukan berupa sapaan dasar dari pengguna. Sesuai dengan mekanisme logika percakapan, jika pengguna mengirimkan sapaan kasual seperti "hi", sistem akan memprosesnya dan membalas dengan kalimat ramah yang menyertakan *Nickname* pengguna secara dinamis. Skenario ini mengonfirmasi bahwa koneksi awal antara antarmuka dan backend telah berjalan dengan baik sebelum pengguna masuk ke fitur utama ekstraksi informasi.

C. Pembaruan Profil dan Nama Pengguna



Gambar 4.44 Profil dan Nickname Berubah

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

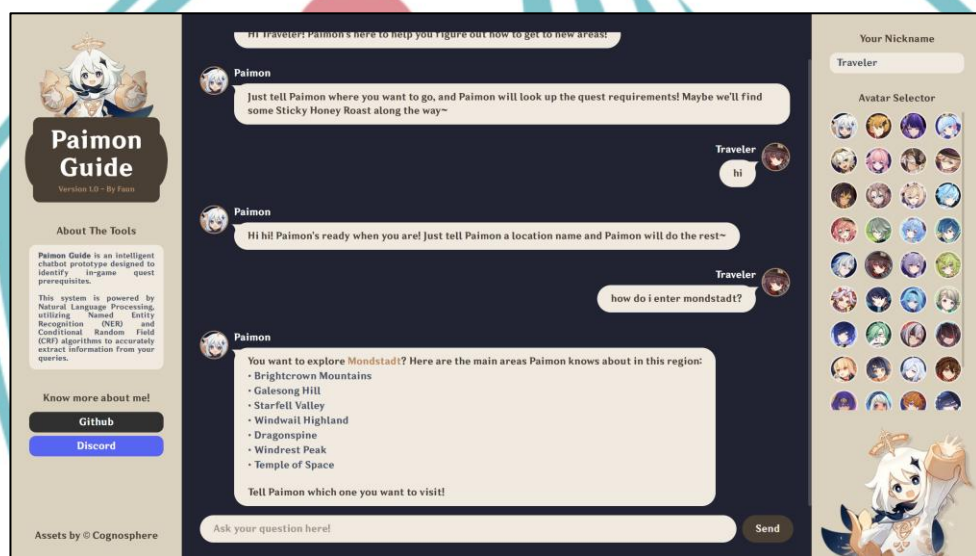


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.44 menampilkan pembaruan antarmuka setelah pengguna melakukan penyesuaian profil pada panel pengaturan. Jika pengguna mengubah identitas *Nickname* (contoh: menjadi "Traveler") beserta ikon avatar, sistem akan langsung memproses perubahan state tersebut secara real-time. Pengguna dapat secara visual mengonfirmasi bahwa preferensi profil mereka telah berhasil tersimpan dan diterapkan pada setiap gelembung pesan (*chat bubble*) yang dikirimkan selanjutnya.

D. Pencarian Berdasarkan Nama Region



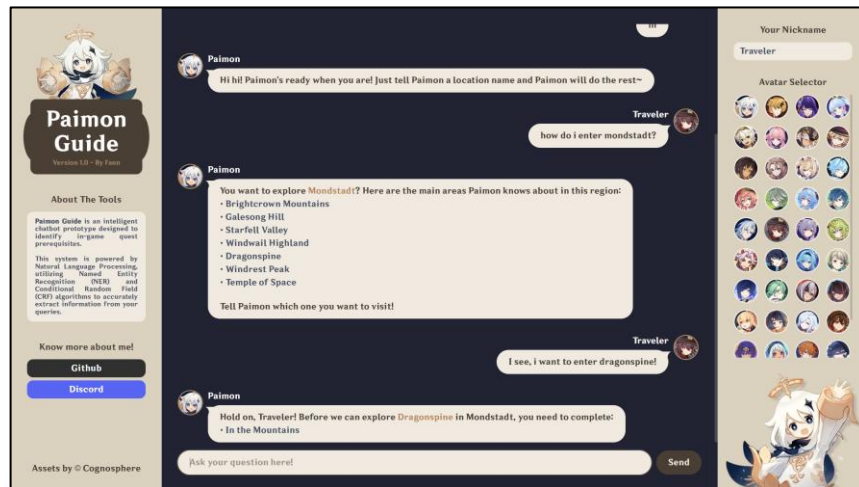
Gambar 4.45 User Mencari Region

Gambar 4.45 mengilustrasikan respons sistem ketika entitas lokasi yang dimasukkan oleh pengguna teridentifikasi sebagai tingkat *Region*. Jika model CRF berhasil mengekstrak entitas yang merujuk pada *Region* (seperti "Mondstadt"), sistem *backend* Laravel akan melakukan *query* ke *database* (tabel *area_prequisites*) dan mengembalikan respons berupa daftar seluruh area yang tergabung di dalam *Region* tersebut. Melalui interaksi ini, pengguna diarahkan untuk mengetikkan nama area yang lebih spesifik guna mendapatkan detail prasyarat *quest*.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

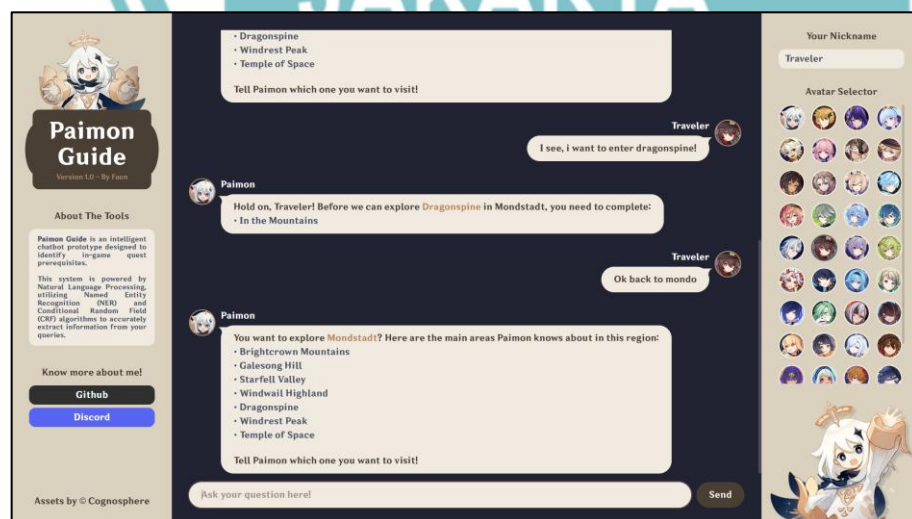
E. Pencarian Berdasarkan Nama Area Spesifik



Gambar 4.46 User Mencari Area Spesifik

Gambar 4.46 memperlihatkan hasil penelusuran sistem ketika menerima masukan pencarian entitas yang spesifik. Apabila algoritma mendeteksi struktur kalimat yang mengandung nama Area (seperti "Dragonspine"), sistem akan secara otomatis mengeksekusi pencarian *knowledge base* pada basis data dan menampilkan informasi detail mengenai prasyarat quest dari area terkait. Pada skenario ini, pengguna berhasil memperoleh panduan otomatis yang menyatakan bahwa mereka harus menyelesaikan misi prasyarat "In the Mountains".

F. Penanganan Input Menggunakan Nama Alias (Singkatan)



Gambar 4.47 User Menanyakan Dengan Istilah

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

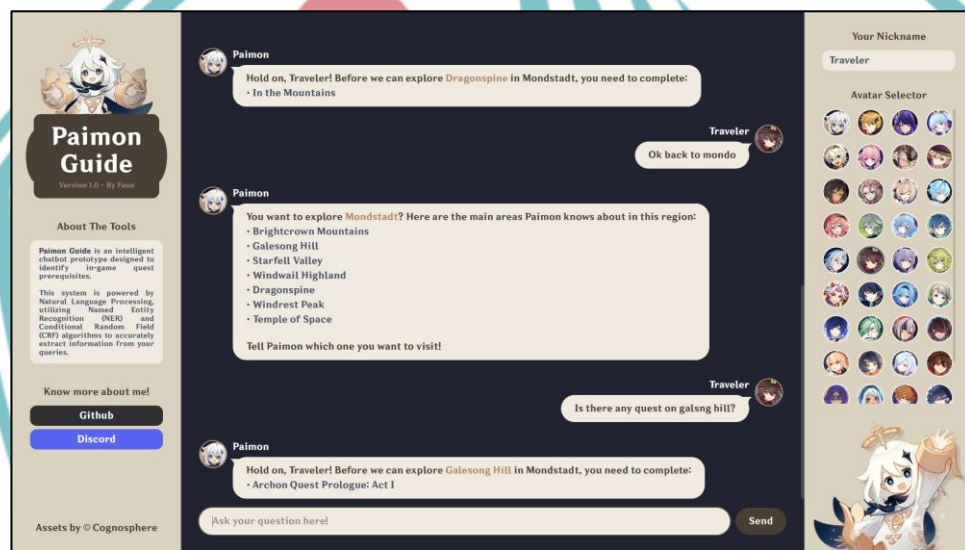


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.47 menyoroti kapabilitas sistem *Entity Resolution* dalam menangani *input* bahasa komunitas atau istilah informal. Apabila pengguna mengetikkan kata ganti seperti "mondo", *chatbot* akan secara cerdas memetakan entitas tersebut menggunakan kamus alias (ALIAS_MAP) dan menormalisasikannya menjadi entitas baku "Mondstadt". Berkat penanganan ini, pengguna tetap akan menerima daftar area yang valid tanpa sistem mengalami disorientasi informasi.

G. Penanganan Masalah Kesalahan Ejaan (Typo)



Gambar 4.48 User Menanyakan Nama Area Dengan Typo

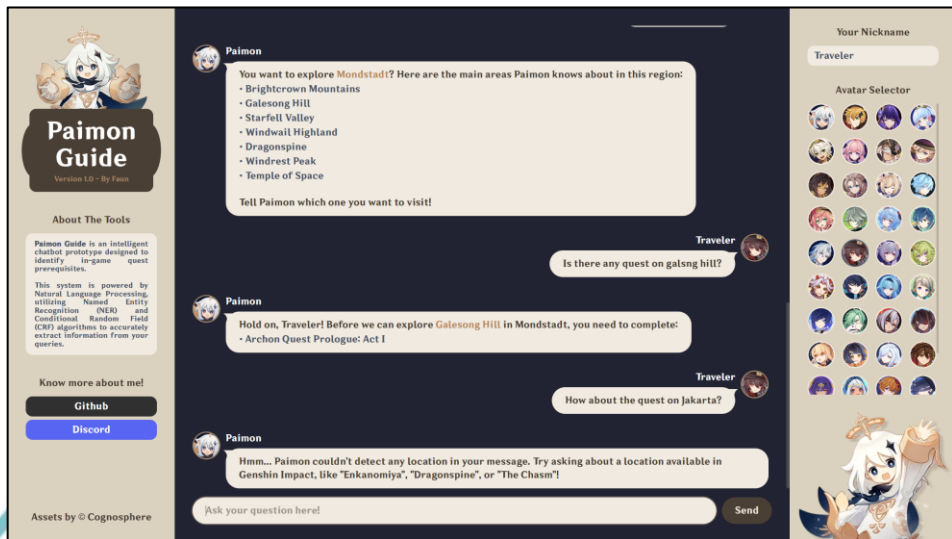
Gambar 4.48 mendemonstrasikan *fault-tolerance* model ketika menghadapi pesan yang mengandung kesalahan ketik (*typo*). Berkat implementasi algoritma *Fuzzy String Matching (Levenshtein Distance)* pada tahap inferensi, masukan pengguna yang keliru secara ejaan seperti "galsng hill" tetap dapat dievaluasi, dikalkulasi jarak kesalahannya, dan diekstrak secara akurat menjadi entitas baku "Galesong Hill". Mekanisme ini memastikan sistem tetap responsif memberikan data prasyarat quest tanpa mengharuskan pengguna mengetik ulang pertanyaan secara sempurna.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

G. Penanganan Pesan Kesalahan



Gambar 4.49 User memasukkan area diluar Database

Gambar 4.49 menunjukkan mekanisme *error handling* yang dijalankan sistem ketika gagal memvalidasi entitas lokasi. Jika pengguna menginputkan nama lokasi yang tidak memiliki kecocokan di dalam *knowledge base* permainan Genshin Impact (seperti "jakarta"), *chatbot* akan mengembalikan respons *error message* dengan status HTTP 404 (*Not Found*) yang menyatakan bahwa lokasi tidak dapat dikenali. Hal ini berfungsi sebagai instruksi validasi mundur yang mengarahkan pengguna untuk memeriksa kembali ejaannya atau memasukkan lokasi spesifik yang hanya ada di dalam game.

4.5 Pengujian Sistem

4.5.1 Deskripsi Pengujian

A. Pengujian *System Usability Scale* (SUS)

Pengujian *System Usability Scale* (SUS) merupakan alat yang dapat diandalkan untuk mengevaluasi kegunaan berbagai produk perangkat lunak melalui metode yang cepat dan sederhana. Metode ini menggunakan kuesioner berisi sepuluh pertanyaan dengan skala Likert lima poin yang menggabungkan pernyataan positif dan negatif untuk memberikan skor



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

numerik terkait tingkat kepuasan dan penerimaan pengguna terhadap sistem tersebut.

B. Pengujian *Black Box*

Pengujian *Black Box* merupakan cara untuk memeriksa fungsi perangkat lunak hanya dengan melihat cara kerjanya dari luar, tanpa perlu melihat kode yang membuatnya berfungsi. Metode pengujian ini secara spesifik berfokus pada pengecekan kesesuaian antara data input dan output guna memastikan bahwa sistem telah bekerja sesuai dengan rencana.

4.5.2 Prosedur Pengujian

A. Prosedur Pengujian *System Usability Scale* (SUS)

Prosedur pengujian *System Usability Scale* dilaksanakan dengan melibatkan responden dari kalangan pemain Genshin Impact sebagai *end-user*. Pada tahap ini, responden diberikan instrumen kuesioner yang memuat 10 pertanyaan pengujian. Sepuluh pertanyaan tersebut secara spesifik dikonstruksi ke dalam dua klasifikasi, yakni lima buah pernyataan bersudut pandang positif dan lima buah pernyataan bersudut pandang negatif.

Proses evaluasi kuesioner diukur menggunakan format penilaian skala lima poin, dengan ketentuan nilai 1 merepresentasikan indikator "Sangat Tidak Setuju" hingga nilai 5 yang menunjukkan indikator "Sangat Setuju". Adapun proses pendistribusian pertanyaan dan pengumpulan respons pengguna tersebut difasilitasi secara daring dengan memanfaatkan layanan pada *platform* Google Forms.

Tabel 7. Kerangka *System Usability Scale*

Kode	Pertanyaan	Sangat Tidak Setuju	Tidak Setuju	Ragu-ragu	Setuju	Sangat Setuju
SUS-1	Saya akan sering menggunakan chatbot ini.					



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SUS-2	Saya merasa chatbot ini terlalu rumit.					
SUS-3	Saya merasa chatbot ini mudah digunakan.					
SUS-4	Saya butuh bantuan teknis untuk menggunakannya.					
SUS-5	Saya merasa fitur-fitur di chatbot ini berjalan dengan baik.					
SUS-6	Saya menemukan hal-hal yang tidak konsisten di chatbot ini.					
SUS-7	Saya yakin orang lain akan cepat paham cara menggunakannya.					
SUS-8	Saya merasa chatbot ini sangat merepotkan.					
SUS-9	Saya merasa yakin dan nyaman saat menggunakannya.					
SUS-10	Saya harus banyak belajar dulu sebelum bisa menggunakannya.					

B. Prosedur Pengujian *Black Box*

Prosedur pengujian *Black Box* dilaksanakan untuk memvalidasi fungsionalitas sistem secara keseluruhan dengan cara mengevaluasi kesesuaian antara nilai masukan (*input*) dan keluaran (*output*) yang dihasilkan oleh aplikasi situs web. Pada tahap ini, pengujian difokuskan murni pada respons antarmuka chatbot tanpa perlu menginspeksi atau memperhatikan detail struktur kode program yang bekerja di latar belakang sistem.

Proses evaluasi dilakukan dengan menyusun serangkaian skenario uji coba fungsional, seperti pengguna memasukkan pesan ke sistem, untuk kemudian mengamati apakah hasil atau respons yang ditampilkan berjalan sesuai dengan harapan dan kebutuhan yang telah dirancang. Seluruh hasil dari skenario pengujian tersebut selanjutnya akan direkam ke dalam bentuk tabel



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

pengujian untuk menetapkan status keberhasilannya (seperti status "Valid") guna memastikan sistem beroperasi dengan semestinya.

Tabel 8. Kerangka Pengujian *Black Box*

ID	Skenario Pengujian	Data Input (Masukan)	Hasil yang Diharapkan (<i>Expected Output</i>)	Hasil Aktual (<i>Actual Output</i>)	Status
TC-01	Mengakses halaman awal web aplikasi	Mengunjungi URL website	Sistem menampilkan halaman beranda dengan panel obrolan, sapaan otomatis pertama, dan menu profil.		
TC-02	Mengirimkan sapaan kasual ke chatbot	Mengetik: "hi"	Sistem membalas dengan kalimat sapaan yang menyertakan Nickname pengguna secara dinamis.		
TC-03	Mengubah Nickname pada menu profil	Mengisi nama: "Traveler" lalu menyimpan	Sistem memperbarui identitas profil pengguna dan menerapkannya pada chat bubble secara real-time.		
TC-04	Melakukan pencarian menggunakan nama tingkat Region	Mengetik: "how to unlock mondstadt"	Sistem mendeteksi entitas Region dan merespons dengan menampilkan daftar area yang ada di dalam Mondstadt.		
TC-05	Melakukan pencarian menggunakan nama spesifik tingkat Area	Mengetik: "where is dragonspine"	Sistem mendeteksi entitas Area dan membalas dengan informasi prasyarat quest		



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

TC-06	Melakukan pencarian menggunakan nama alias/informal	Mengetik: "quest for mondo"	Sistem berhasil me-resolusi alias "mondo" menjadi "Mondstadt" dan menampilkan daftar areanya.		
TC-07	Melakukan pencarian dengan input yang salah eja (typo)	Mengetik: "how to open galsng hill"	Fungsi Fuzzy Matching memperbaiki kata menjadi "Galesong Hill" dan menampilkan informasi quest-nya.		
TC-08	Menginputkan nama lokasi fiktif di luar konteks game	Mengetik: "how about going to jakarta"	Sistem mendeteksi bahwa lokasi tidak ada di basis data dan merespons dengan pesan validasi/error HTTP 404.		

4.5.3 Data Hasil Pengujian

A. Pengujian *System Usability Scale* (SUS)

Pelaksanaan pengujian *System Usability Scale* pada penelitian ini melibatkan 10 orang responden yang merepresentasikan pemain Genshin Impact sebagai target pengguna dari sistem chatbot Paimon Guide. Sebelum mengoperasikan sistem, para responden terlebih dahulu diberikan pemaparan singkat terkait gambaran umum dan fungsionalitas aplikasi. Setelah mendapatkan pemahaman dasar tersebut, responden diarahkan untuk mencoba langsung fitur-fitur pencarian informasi prasyarat quest pada sistem, untuk kemudian memberikan penilaian evaluasi kebergunaan aplikasi menggunakan instrumen kuesioner yang didistribusikan melalui Google Forms.

Tabel 9. Hasil Pengujian *System Usability Scale*

Kode Responden	Pertanyaan									
	SUS -1	SUS -2	SUS -3	SUS -4	SUS -5	SUS -6	SUS -7	SUS -8	SUS -9	SUS -10
R-1	4	2	5	1	5	1	4	1	5	1
R-2	4	1	4	2	4	4	5	1	4	1
R-3	3	2	4	1	4	3	4	2	4	1
R-4	4	1	5	3	4	3	4	2	4	1
R-5	3	1	5	2	4	2	5	2	4	1
R-6	3	1	5	1	4	4	5	1	5	1
R-7	3	2	4	1	5	5	4	1	4	1
R-8	4	1	5	1	4	1	5	1	5	1
R-9	5	1	5	3	4	2	4	1	4	2
R-10	4	2	5	1	4	2	4	2	5	1

Setelah data penilaian dari para responden terkumpul, tahapan selanjutnya adalah melakukan kalkulasi skor *System Usability Scale*. Bobot nilai untuk setiap butir pertanyaan ganjil dihitung dengan cara mengurangi nilai jawaban responden dengan angka 1. Sebaliknya, bobot nilai untuk butir pertanyaan genap diperoleh dengan cara mengurangi angka 5 dengan nilai jawaban yang diberikan oleh responden. Dari total nilai yang didapat, kemudian dikalikan dengan 2.5 untuk mendapatkan Skor Akhir SUS.

Tabel 10. Perhitungan Skor *System Usability Scale*

Kode Responden	Nilai SUS Ganjil	Nilai SUS Genap	Total Nilai	Skor SUS
R-1	18	19	37	92,5
R-2	16	16	32	80
R-3	14	16	30	75
R-4	16	15	31	77,5
R-5	16	17	33	82,5
R-6	17	17	34	85



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

R-7	15	15	30	75
R-8	18	20	38	95
R-9	17	16	33	82,5
R-10	17	17	34	85
Skor SUS Rata-Rata				83

Dari Skor SUS rata-rata yang telah didapat, dikonversi menjadi Rating dengan bentuk tabel seperti dibawah:

Tabel 11. *Grade & Rating* SUS Score

SUS Score	Grade	Rating
>80,3	A	<i>Excellent</i>
68 – 80,2	B	<i>Good</i>
67	C	<i>Okay</i>
51 – 66	D	<i>Poor</i>
<51	F	<i>Awful</i>

Dari tabel 11. mengenai *Grade & Rating* skor SUS, maka sistem chatbot Paimon Guide yang mencapai skor SUS 83 dinilai sebagai *Grade* “A” dengan rating “*Excellent*”. Data tersebut membuktikan bahwa fungsionalitas dan antarmuka interaksi pada sistem sudah sangat layak, mudah dipahami, serta memiliki *acceptability* yang tinggi oleh pengguna.

B. Pengujian *Black Box*

Tabel 12. Data Hasil Pengujian *Black Box*

ID	Skenario Pengujian	Data Input (Masukan)	Hasil yang Diharapkan (Expected Output)	Hasil Aktual (Actual Output)	Status
TC-01	Mengakses halaman awal web aplikasi	Mengunjungi URL website	Sistem menampilkan halaman beranda dengan panel obrolan, sapaan	Berhasil menampilkan beranda dan pesan chatbot.	Valid



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

			otomatis pertama, dan menu profil.		
TC-02	Mengirimkan sapaan kasual ke chatbot	Menetik: "hi"	Sistem membalas dengan kalimat sapaan yang menyertakan Nickname pengguna secara dinamis.	Sistem merespons dengan sapaan dan memanggil dengan Nickname pengguna.	Valid
TC-03	Mengubah Nickname pada menu profil	Mengisi nama: "Traveler" lalu menyimpan	Sistem memperbarui identitas profil pengguna dan menerapkannya pada chat bubble secara real-time.	Nama profil berubah menjadi "Traveler" dan pada tampilan pesan.	Valid
TC-04	Melakukan pencarian menggunakan nama tingkat Region	Menetik: "how to unlock mondstadt"	Sistem mendeteksi entitas Region dan merespons dengan menampilkan daftar area yang ada di dalam Mondstadt.	Sistem menampilkan list area valid yang merupakan bagian dari Mondstadt.	Valid
TC-05	Melakukan pencarian menggunakan nama spesifik tingkat Area	Menetik: "where is dragonspine"	Sistem mendeteksi entitas area dan membalas dengan informasi prasyarat quest	Sistem langsung menampilkan data prasyarat quest "In the Mountains" secara akurat.	Valid
TC-06	Melakukan pencarian menggunakan nama alias/informal	Menetik: "quest for mondo"	Sistem berhasil merevisi alias "mondo" menjadi "Mondstadt" dan menampilkan daftar areanya.	Kata "mondo" berhasil dikonversi dan sistem merespons dengan daftar area Mondstadt.	Valid
TC-07	Melakukan pencarian dengan input yang salah eja (typo)	Menetik: "how to open galsng hill"	Fungsi Fuzzy Matching memperbaiki kata menjadi "Galesong Hill" dan	Typo terdeteksi, kata dikoreksi otomatis, dan detail quest Galesong Hill	Valid



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

			menampilkan informasi quest-nya.	berhasil ditampilkan.	
TC-08	Menginputkan nama lokasi fiktif di luar konteks game	Mengetik: "how about going to jakarta"	Sistem mendeteksi bahwa lokasi tidak ada di basis data dan merespons dengan pesan validasi/error HTTP 404.	Sistem mengeluarkan pesan peringatan bahwa lokasi "jakarta" tidak dikenali oleh Paimon.	Valid

4.5.4 Analisis Hasil Pengujian

1. Analisis Hasil *System Usability Scale* (SUS)

Perolehan nilai rata-rata sebesar 83 di tabel 10. menunjukkan bahwa sistem chatbot Paimon Guide masuk ke dalam kategori Grade “A” dengan predikat “Excellent” berdasarkan tabel 11. Hasil kalkulasi tersebut mengindikasikan bahwa rancangan fungsionalitas serta antarmuka interaksi dari sistem ini telah teruji sangat layak. Perolehan skor ini merepresentasikan bahwa perangkat lunak yang dikembangkan tidak hanya mudah dipahami dan dioperasikan, tetapi juga memiliki tingkat penerimaan (*acceptability*) yang sangat tinggi dari sisi pengguna akhir.

Keunggulan utama dari *chatbot* ini terletak pada aspek kemudahan pengoperasian secara fundamental. Mayoritas responden memberikan skor maksimal (skala 4 dan 5) pada indikator kemudahan penggunaan (SUS-3) dan mereka tidak harus banyak belajar terlebih dahulu sebelum bisa menggunakan sistem (SUS-10). Keberhasilan desain ini juga didukung oleh rendahnya penilaian terhadap indikator kerumitan (SUS-2) dan anggapan bahwa sistem merepotkan (SUS-8). Namun sistem ini masih memiliki kekurangan minor yang terpusat pada aspek konsistensi. Hal ini teridentifikasi dari adanya beberapa responden yang memberikan nilai menengah hingga tinggi (skala 3 hingga 5) pada pernyataan terkait penemuan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

hal-hal yang tidak konsisten di dalam *chatbot* (SUS-6). Inkonsistensi tersebut kemungkinan menjadi faktor penyebab mengapa segelintir pengguna merasa masih sedikit membutuhkan bantuan teknis (SUS-4) pada saat-saat tertentu. Dari feedback balik kualitatif responden, inkonsistensi tersebut utamanya disebabkan oleh tingkat sensitivitas algoritma *typo checker* yang mempengaruhi akurasi jawaban.

2. **Analisis Hasil Pengujian *Black Box***

Merujuk pada rekapitulasi pengujian *Black Box* yang disajikan pada Tabel 12, seluruh skenario uji terbukti dapat tereksekusi dengan baik dan menghasilkan keluaran yang valid sesuai dengan spesifikasi rancangan sistem. Tingkat keberhasilan fungsional ini mengindikasikan bahwa *chatbot* telah mencapai persentase kelayakan operasional sebesar 100%. Perolehan persentase mutlak tersebut dievaluasi secara eksklusif berdasarkan parameter skenario pengujian yang telah didefinisikan sebelumnya di dalam tabel. Keberhasilan pemenuhan seluruh skenario uji ini sekaligus memberikan jaminan teknis bahwa fitur-fitur utama sistem telah terimplementasi dengan stabil dan terbebas dari kecacatan program (*bug*).

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB 5

PENUTUP

5.1 Kesimpulan

Penelitian ini telah berhasil mengimplementasikan sistem *chatbot* untuk informasi game dengan algoritma *Conditional Random Field*, dimana hasil yang dipadukan dengan teknik augmentasi data sangat efektif dalam menyelesaikan tugas ekstraksi entitas prasyarat *quest*. Penelitian ini menunjukkan adanya perbedaan performa dan stabilitas yang signifikan antara pemodelan yang menggunakan dataset mentah dibandingkan dengan dataset yang telah di augmentasi. Konfigurasi pembagian data uji 80:20 dengan augmentasi mencatatkan performa tertinggi secara konsisten pada seluruh metrik pengujian. Temuan ini sekaligus menjawab rumusan masalah utama penelitian dengan mengonfirmasi bahwa pendekatan *Named Entity Recognition* mampu mendeteksi entitas lokasi permainan secara presisi.

Keberhasilan implementasi pemodelan ini ke dalam sistem *chatbot* didukung oleh beberapa faktor penentu:

- Kompensasi Keterbatasan Data Mentah: Penerapan augmentasi data terbukti menjadi strategi yang krusial untuk mengatasi fluktuasi kinerja model. Berbeda dengan model tanpa augmentasi yang hanya menghasilkan nilai *recall* keseluruhan sebesar 13,33% (pada skenario 80:20), pemodelan dengan augmentasi pada konfigurasi data optimal 80:20 berhasil meningkatkan nilai *recall* secara signifikan menjadi 99,58%. Model yang dikembangkan berhasil menyimulasikan variasi dan kerumitan bahasa pemain yang lebih luas secara efektif, sehingga pada akhirnya sukses mencetak tingkat *accuracy* maksimal mencapai 99,84% dan nilai *F1-Score* sebesar 99,37%.
- Kemampuan Generalisasi Leksikal: Evaluasi pada *learning curve* mengindikasikan adanya fenomena *overfitting* ringan pada model di fase awal hingga menengah. Meskipun model menghafal data latih secara absolut, hal tersebut sama sekali tidak merusak performa secara keseluruhan. Model terbukti tetap mempertahankan kemampuan adaptasi dan generalisasi yang sangat tinggi terhadap kueri baru. Kinerja optimal ini dipengaruhi oleh



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

pemanfaatan korpus data latih yang diformulasikan secara terarah dari *knowledge base*. Sifat data yang homogen dan terstruktur tersebut memfasilitasi algoritma CRF untuk mengenali batas-batas entitas dengan sangat cepat sejak awal fase pelatihan tanpa mengalami *underfitting*.

- Penyelarasan Arsitektur Model dan Usabilitas: Implementasi teknis model ke dalam aplikasi tidak mengurangi kualitas pengalaman *end-user*. Selain terbukti fungsional sepenuhnya melalui pengujian Black Box, integrasi model NLP ini berhasil disajikan dalam bentuk *user interface* interaktif yang mencatatkan skor *System Usability Scale* sebesar 83 (Kategori *Excellent*). Hal ini menjelaskan bahwa kompleksitas algoritma di balik layar berhasil diterjemahkan menjadi interaksi pengguna yang intuitif dan praktis.

Dengan demikian, dapat disimpulkan bahwa analisis teks menggunakan CRF berbasis data yang di augmentasi merupakan solusi yang sangat andal untuk mengotomatisasi pencarian prasyarat quest dalam format percakapan.

5.2 Saran

Berdasarkan hasil analisis penelitian ini, terdapat peluang pengembangan lebih lanjut yang dapat dieksplorasi di masa depan untuk menyempurnakan keandalan sistem:

- Pengembangan selanjutnya perlu mengeksplorasi algoritma koreksi ejaan leksikal yang lebih dinamis terkait inkonsistensi jawaban yang dipicu oleh keterbatasan *typo checker*. Peningkatan sensitivitas terhadap kesalahan penulisan (*tolerance*) akan memastikan agar akurasi pemrosesan NER tetap kokoh meskipun masukan pengguna tidak memenuhi standar penulisan baku.
- Penelitian selanjutnya disarankan untuk memperluas cakupan pelatihan menggunakan dataset percakapan yang lebih heterogen mengingat tingginya metrik evaluasi model sangat terbantu oleh kondisi data leksikon yang spesifik dan homogen. Memasukkan struktur sintaksis yang lebih kompleks dan tidak berpola akan menguji ketahanan model secara lebih komprehensif dalam skenario percakapan dunia nyata yang tidak terprediksi.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Arimbi, Y.D., Kartinah, D. dan Della, A.N.W. (2022) “Rancangan Sistem Informasi Kost Putri Malika Berbasis Website Menggunakan Framework Laravel Dan Mysql,” *Jurnal Ilmiah Multidisiplin*, 1(03), hal. 93–103. Tersedia pada: <https://doi.org/10.56127/jukim.v1i03.201>.
- Azizi, A. *et al.* (2024) “Pengembangan Dashboard Admin Bukupedia,” *Jurnal Teknik Indonesia*, 3(1). Tersedia pada: <https://doi.org/10.58860/jti.v3i1>.
- Budi, I. dan Suryono, R.R. (2023) “Application of named entity recognition method for Indonesian datasets: a review,” *Bulletin of Electrical Engineering and Informatics*, 12(2), hal. 969–978. Tersedia pada: <https://doi.org/10.11591/eei.v12i2.4529>.
- Darwansah, W.M., Arifiyanti, A.A. dan Hadiwiyan, R. (2025) “Classification and Mapping of Online Gambling Based on News Articles Using NER and SVM,” 8(2), hal. 788–797.
- Fadila, N. *et al.* (2024) “Perancangan Prototype User Interface Sistem Informasi Pusat Data Dokumen Menggunakan Figma,” *Jurnal Komputer, Informasi dan Teknologi*, 4(1), hal. 1. Tersedia pada: <https://penerbitadm.pubmedia.id/index.php/KOMITEK>.
- Genshin Impact Wiki (2025) *World Quest*. Tersedia pada: https://genshin-impact.fandom.com/wiki/World_Quest?veaction=edit (Diakses: 21 April 2026).
- Genshin Impact Wiki (2026a) *Exploration*. Tersedia pada: <https://genshin-impact.fandom.com/wiki/Exploration> (Diakses: 21 April 2025).
- Genshin Impact Wiki (2026b) *Genshin Impact*. Tersedia pada: https://genshin-impact.fandom.com/wiki/Genshin_Impact (Diakses: 21 April 2026).
- Hafizhah, A., Theresiawati, T. dan Warsuta, B. (2022) “Sistem Informasi Pariwisata Badan Usaha Milik Desa Tirta Sejahtera Pada Masa Pandemi Berbasis Website (Studi Kasus: Wisata Desa Pluneng),” *Informatik : Jurnal*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Ilmu Komputer, 18(2), hal. 113. Tersedia pada:
<https://doi.org/10.52958/iftk.v18i2.4638>.

Handrianto, Y. dan Sanjaya, B. (2020) “Model Waterfall Dalam Rancang Bangun Sistem Informasi Pemesanan Produk Dan Outlet Berbasis Web,” *Jurnal Inovasi Informatika*, 5(2), hal. 153–160. Tersedia pada:
<https://doi.org/10.51170/jii.v5i2.66>.

Hasanah, N. dan Indriawan, M.N. (2021) “Rancangan Aplikasi Batam Travel Menggunakan Metode Software Development Life Cycle (SDLC),” *CoMBInES - Conference on Management, Business, Innovation, Education and Social Sciences*, 1(1), hal. 925–938. Tersedia pada:
<https://journal.uib.ac.id/index.php/combines/article/view/4524>.

Illescas-Manzano, M.D. *et al.* (2021) “Implementation of chatbot in online commerce, and open innovation,” *Journal of Open Innovation: Technology, Market, and Complexity*, 7(2). Tersedia pada:
<https://doi.org/10.3390/joitmc7020125>.

Jaariyah, N. dan Rainarli, E. (2017) “CONDITIONAL RANDOM FIELDS UNTUK PENGENALAN ENTITAS BERNAMA PADA TEKS BAHASA INDONESIA,” *Komputa : Jurnal Ilmiah Komputer dan Informatika*, 6(1), hal. 29–34. Tersedia pada: <https://doi.org/10.34010/komputa.v6i1.2474>.

Jehangir, B., Radhakrishnan, S. dan Agarwal, R. (2023) “A survey on Named Entity Recognition — datasets, tools, and methodologies,” *Natural Language Processing Journal*, 3(10), hal. 100017. Tersedia pada:
<https://doi.org/10.1016/j.nlp.2023.100017>.

Karo Karo, I.M., Dewi, S. dan Syahrin, A. (2025) “Ekstraksi Informasi Bencana Banjir Dari Berita Online Berbasis Named Entity Recognition,” *Multinetics*, 11(1), hal. 35–42. Tersedia pada:
<https://doi.org/10.32722/multinetics.v11i1.7499>.

Khan, W. *et al.* (2022) “Urdu Named Entity Recognition Using Conditional Random Fields,” *Applied Sciences (Switzerland)*, 12(13), hal. 1–18. Tersedia pada: <https://doi.org/10.3390/app12136391>.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kosim, M.A., Aji, S.R. dan Darwis, M. (2022) “Pengujian Usability Aplikasi Pedulilindungi Dengan Metode System Usability Scale (Sus),” *Jurnal Sistem Informasi dan Sains Teknologi*, 4(2), hal. 1–7. Tersedia pada: <https://doi.org/10.31326/sistek.v4i2.1326>.

Kurniawan, D. dan Yasir, M. (2022) “Optimization Sentimen Analysis using CRISP-DM and Naive Bayes Methods Implemented on Social Media,” *Cyberspace: Jurnal Pendidikan Teknologi Informasi*, 6(2), hal. 74. Tersedia pada: <https://doi.org/10.22373/cj.v6i2.12793>.

Ma, C. (2025) “Research on the Transformation Mechanism and Influencing Factors of Open-World Game Design to Players ’ Free Labor--Based on the analysis of Digital Labor and Player Labor Theory,” 220. Tersedia pada: <https://doi.org/https://doi.org/10.1051/shsconf/202522004028>.

Muhamad Malik Matin, I. (2023) “Hyperparameter Tuning Menggunakan GridsearchCV pada Random Forest untuk Deteksi Malware,” *Multinetics*, 9(1), hal. 43–50. Tersedia pada: <https://doi.org/10.32722/multinetics.v9i1.5578>.

Narulita, S., Nugroho, A. dan Abdillah, M.Z. (2024) “Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS) Universitas Nasional Karangturi Semarang , Indonesia (deskripsi) dan perancangan sistem , khususnya pada pemrogr,” (3), hal. 244–256.

Potočár, M. dan Kvet, M. (2023) “Comparison of Unigram, HMM, CRF and Brill’s Part-of-Speech Taggers Available in NLTK Library,” *Conference of Open Innovation Association, FRUCT*, 2023-May, hal. 226–235. Tersedia pada: <https://doi.org/10.23919/FRUCT58615.2023.10143061>.

Santoso, J. *et al.* (2020) “Hybrid conditional random fields and k-means for named entity recognition on indonesian news documents,” *International Journal of Intelligent Engineering and Systems*, 13(3), hal. 233–245. Tersedia pada: <https://doi.org/10.22266/IJIES2020.0630.22>.

Santoso, J.T. dan Yan, S. (2024) “A Hybrid Approach to Typo Correction in



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Indonesian Documents Using Levenshtein Distance,” 3(2), hal. 151–168.
Tersedia pada: <https://doi.org/10.51903/jtie.v3i2.184>.

Setiyani, L. dan Tjandra, E. (2021) “Analisis Kebutuhan Fungsional Aplikasi Penanganan Keluhan Mahasiswa Studi Kasus: Stmik Rosma Karawang,” *Jurnal Inovasi Pendidikan dan Teknologi Informasi (JIPTI)*, 2(1), hal. 8–17. Tersedia pada: <https://doi.org/10.52060/pti.v2i01.465>.

Simorangkir, A. *et al.* (2024) “Pemodelan Chatbot Rekomendasi Hotel Dengan Menggunakan Natural Language Processing,” *Simtek: jurnal sistem informasi dan teknik komputer*, 9(1), hal. 46–50. Tersedia pada: <https://doi.org/10.51876/simtek.v9i1.371>.

Solikhah, E.W. dan Ngetich, B.K. (2025) “The Impact of Using Chatbots in Improving Customer Service in The E-commerce Industry,” *ProBisnis: Jurnal Manajemen*, 16(5), hal. 962–970.

Suraya dan Sholeh, M. (2022) “Designing and Implementing a Database for Thesis Data Management by Using the Python Flask Framework,” *International Journal of Engineering, Science and Information Technology*, 2(1), hal. 9–14. Tersedia pada: <https://doi.org/10.52088/ijesty.v1i1.197>.

Witriyono, H. dan Aipina, D. (2022) “Pemanfaatan Framework Laravel Dan Framework Bootstrap Pada Pembangunan Aplikasi Penjualan Hijab Berbasis Web,” *Jurnal Media Infotama*, 18(1), hal. 2022.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



DAFTAR RIWAYAT HIDUP

Muhammad Fajrul Falah, lahir di Jakarta, 28 April 2004. Lulus dari SDIT Azzahra Depok tahun 2016, MTsN 3 Jakarta tahun 2019, SMAN 10 Depok tahun 2022, program pendidikan professional CCIT-FTUI (*Center for Computing and Information Technology* Fakultas Teknik Universitas Indonesia) Tahun 2024 dan saat ini sedang menempuh pendidikan Diploma IV Program Studi Teknik Informatika di Politeknik Negeri Jakarta pada tahun 2022.

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

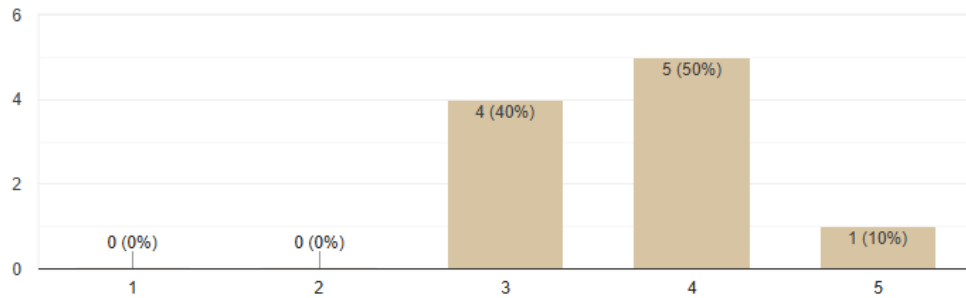
LAMPIRAN

Lampiran 1 Hasil Pertanyaan Kuesioner SUS

Saya akan sering menggunakan *chatbot* ini.

[Copy chart](#)

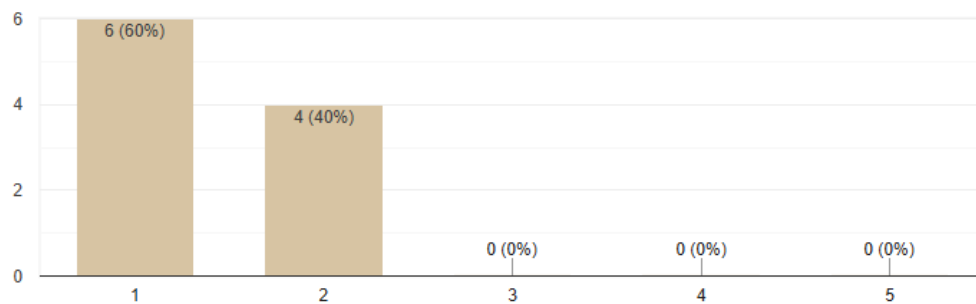
10 responses



Saya merasa *chatbot* ini terlalu rumit.

[Copy chart](#)

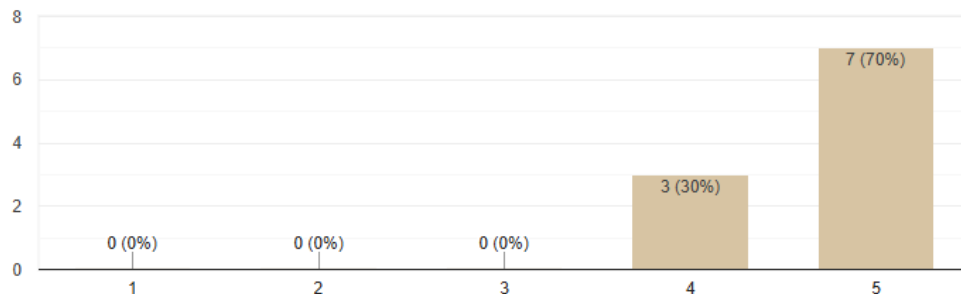
10 responses



Saya merasa *chatbot* ini mudah digunakan.

[Copy chart](#)

10 responses





© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

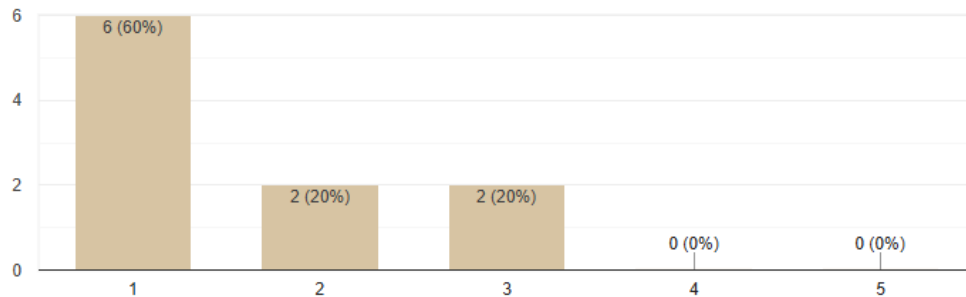
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

(Lanjutan)

Copy chart

Saya butuh bantuan teknis untuk menggunakannya.

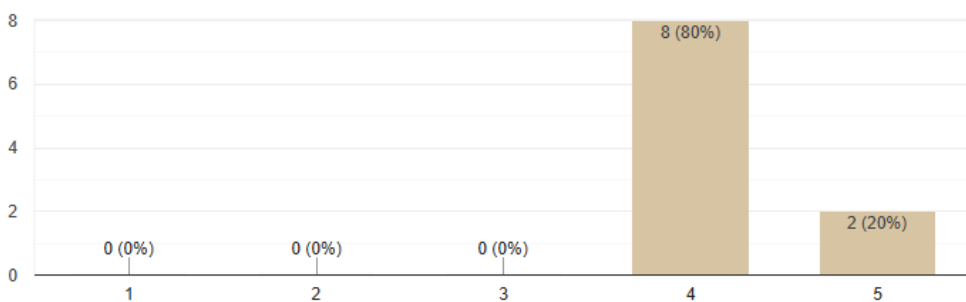
10 responses



Saya merasa fitur-fitur di *chatbot* ini berjalan dengan baik.

Copy chart

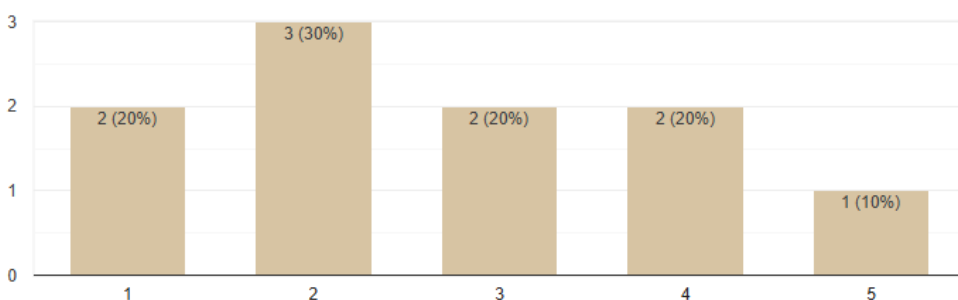
10 responses



Saya menemukan hal-hal yang tidak konsisten di *chatbot* ini.

Copy chart

10 responses





© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

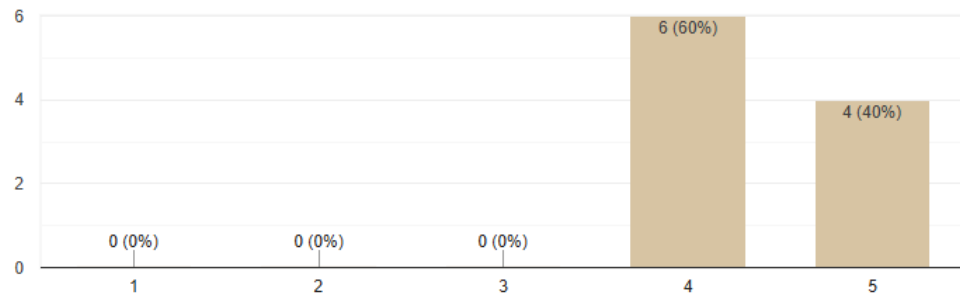
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

(Lanjutan)

Saya yakin orang lain akan cepat paham cara menggunakannya.

[Copy chart](#)

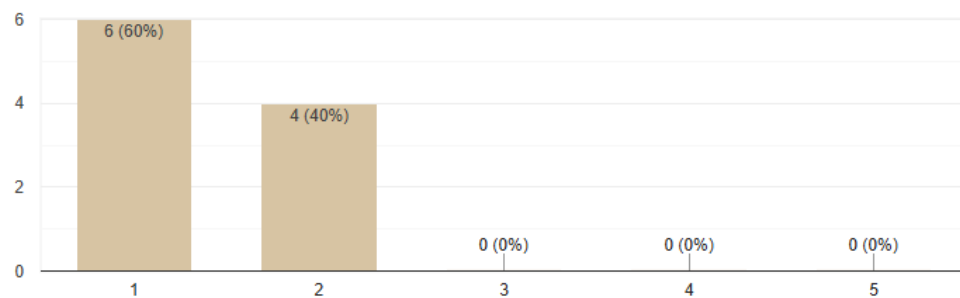
10 responses



Saya merasa *chatbot* ini sangat merepotkan.

[Copy chart](#)

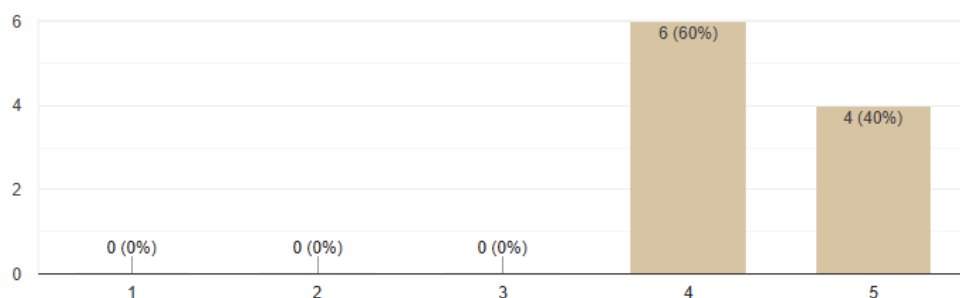
10 responses



Saya merasa yakin dan nyaman saat menggunakannya.

[Copy chart](#)

10 responses





© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

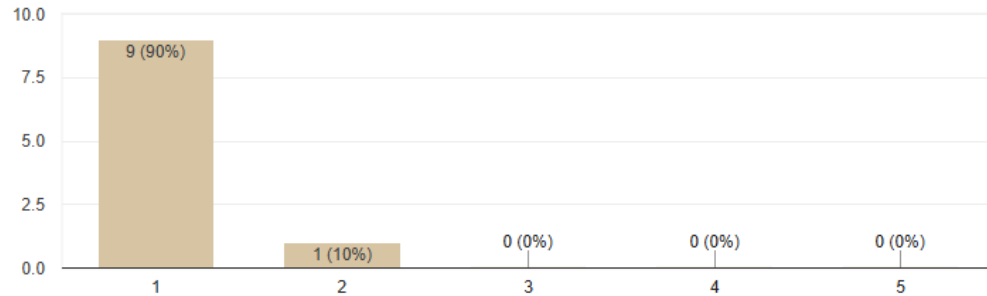
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

(Lanjutan)

Copy chart

Saya harus banyak belajar dulu sebelum bisa menggunakannya.

10 responses





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



KEMENTERIAN PENDIDIKAN TINGGI, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI JAKARTA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
Universitas Indonesia, Jl. Prof. DR. G.A. Siwabessy, Depok 16425
Telp: (021)91274097, Fax : (021) 7863531, (021)7270036 Hunting
Laman :<http://www.pnj.ac.id>, e-mail : tik.pnj@gmail.com

L1

Pada hari ini, Sabtu, Tanggal 25 April 2026, telah dilaksanakan verifikasi dan validasi data hasil penelitian bersama :

Nama : Patriot Antaras Bangsa
Peran : Validator Dataset

Sesi ini dilakukan dalam rangka validasi hasil pelabelan data teks (anotasi) menggunakan format *Inside-Outside-Beginning* (IOB) untuk pengenalan entitas lokasi pada gim Genshin Impact. Dataset ini akan digunakan untuk melatih algoritma *Conditional Random Field* (CRF) dalam penelitian implementasi sistem chatbot. Adapun kegiatan yang dilakukan selama sesi berlangsung dapat dilihat pada poin-poin berikut:

1. Peneliti menjelaskan secara umum ruang lingkup penelitian, pedoman anotasi (pelabelan) data teks, dan batasan entitas lokasi pada gim Genshin Impact guna menyamakan pemahaman antara peneliti dan validator.
2. Validator melakukan pengecekan dan pengujian secara manual terhadap keakuratan hasil pelabelan entitas lokasi (B-LOC, I-LOC, dan O) pada sampel korpus teks bahasa alami pengguna yang diambil dari obrolan *server* Discord Genshin Impact.
3. Validator memberikan penilaian kuantitatif menggunakan skala Likert (1–5) berdasarkan parameter pengujian validitas data latih yang tercantum pada lembar lampiran kedua.

Berdasarkan hasil wawancara dan pengujian tersebut, Validator menyatakan bahwa hasil pelabelan (anotasi) entitas lokasi pada dataset latih dinyatakan **benar** dan sesuai dengan format *Named Entity Recognition* (NER) serta terminologi yang berlaku di dalam gim Genshin Impact.

Demikian Berita Acara ini dibuat dengan sebenarnya untuk dapat digunakan sebagaimana mestinya.

Depok, 25 April 2026

Pihak Pertama (Peneliti)

Muhammad Fajrul Falah
NIM. 2207412019

Pihak Kedua (Validator Dataset)

Patriot Antaras Bangsa
NIM. 24060123130100

(Lanjutan)



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



KEMENTERIAN PENDIDIKAN TINGGI, RISET, DAN TEKNOLOGI
 POLITEKNIK NEGERI JAKARTA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
 Universitas Indonesia, Jl. Prof. DR. G.A. Siwabessy, Depok 16425
 Telp: (021)91274097, Fax : (021) 7863531, (021) 7270036 Hunting
 Laman : <http://www.pnj.ac.id>, e-mail : tik.pnj@gmail.com

L1

LEMBAR PARAMETER PENGUJIAN HASIL VALIDASI DATA

Kode Uji	Pernyataan	Nilai
VAL -1	Kata atau frasa yang diberi label entitas lokasi benar-benar merepresentasikan area, wilayah, atau sebutan spesifik yang ada di dalam gim Genshin Impact (termasuk sebutan atau istilah tidak baku dari komunitas pemain).	Penilaian dengan menggunakan skala likert 1-5. dengan aturan sebagai berikut. - 1 = Sangat Tidak Setuju - 2 = Tidak Setuju - 3 = Cukup Setuju - 4 = Setuju - 5 = Sangat Setuju
VAL-2	Penggunaan label Inside-Outside-Beginning pada setiap token/kata sudah tepat; yaitu awalan kata lokasi ditandai dengan B-LOC, lanjutan kata lokasi ditandai dengan I-LOC, dan kata selain entitas lokasi ditandai dengan O.	
VAL-3	Anotator sudah konsisten dalam melabeli entitas lokasi pada berbagai variasi struktur kalimat, termasuk perlakuan yang tepat terhadap tanda baca, huruf kapital, imbuhan, serta kesalahan ketik (typo)	



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



KEMENTERIAN PENDIDIKAN TINGGI, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI JAKARTA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
Universitas Indonesia, Jl. Prof. DR. G.A. Siwabessy, Depok 16425
Telp: (021)91274097, Fax : (021) 7863531, (021) 7270036 Hunting
Laman : <http://www.pnj.ac.id>, e-mail : tik.pnj@gmail.com

L1

	dari teks asli obrolan Discord.	
--	------------------------------------	--

Kode Uji	Skenario Pengujian	Nilai
VAL-1	Kata atau frasa yang diberi label entitas lokasi benar-benar merepresentasikan area, wilayah, atau sebutan spesifik yang ada di dalam gim Genshin Impact (termasuk sebutan atau istilah tidak baku dari komunitas pemain).	5
VAL-2	Penggunaan label Inside-Outside-Beginning pada setiap token/kata sudah tepat; yaitu awalan kata lokasi ditandai dengan B-LOC, lanjutan kata lokasi ditandai dengan I-LOC, dan kata selain entitas lokasi ditandai dengan O.	5
VAL-3	Anotator sudah konsisten dalam melabeli entitas lokasi pada berbagai variasi struktur kalimat, termasuk perlakuan yang tepat terhadap tanda baca, huruf kapital, imbuhan, serta kesalahan ketik (typo)	5



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



KEMENTERIAN PENDIDIKAN TINGGI, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI JAKARTA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
Universitas Indonesia, Jl. Prof. DR. G.A. Siwabessy, Depok 16425
Telp: (021) 91274097, Fax : (021) 7863531, (021) 7270036 Hunting
Laman : <http://www.pnj.ac.id>, e-mail : tik.pnj@gmail.com

L2

Pada hari ini, Sabtu, Tanggal 25 April 2026, telah diselesaikan verifikasi anotasi untuk keperluan dataset penelitian bersama:

Nama : Raffael Lawrenzier
Peran : Anotator Dataset

Proses anotasi ini dilakukan untuk membangun dataset pengenalan entitas lokasi pada korpus teks bahasa alami dari *server* Discord gim Genshin Impact. Dataset ini akan digunakan untuk melatih algoritma *Conditional Random Field* (CRF) dalam implementasi sistem chatbot. Adapun kegiatan yang telah dilaksanakan selama proses anotasi berlangsung adalah sebagai berikut:

1. Peneliti dan Anotator menyelaraskan pemahaman mengenai *Pedoman Anotasi (Annotation Guideline)* yang mencakup aturan pelabelan format *Inside-Outside-Beginning* (IOB) serta batasan entitas lokasi sesuai dengan terminologi gim Genshin Impact.
2. Anotator telah melakukan pelabelan entitas secara mandiri pada sampel teks kalimat yang diambil dari obrolan pengguna Discord menggunakan label B-LOC untuk awalan lokasi, I-LOC untuk lanjutan lokasi, dan O untuk kata di luar lokasi.

Berdasarkan kegiatan tersebut, Anotator menyatakan bahwa proses pelabelan dataset telah diselesaikan secara objektif sesuai dengan pedoman yang telah disepakati bersama.

Demikian Berita Acara ini dibuat dengan sebenarnya untuk dapat dilampirkan sebagai bukti pelaksanaan tahapan pra-pemrosesan data penelitian.

Depok, 25 April 2026

Pihak Pertama (Peneliti)

Muhammad Fajrul Falah

NIM. 2207412019

Pihak Kedua (Anotator Dataset)

Raffael Lawrenzier

NIM. BI210006