



PROGRES PROYEK AKHIR

**Penerapan *Large Language Model (LLM)* untuk
Rekomendasi Lowongan Pekerjaan Berdasarkan
Analisis CV Pengguna**

ARDIANSYAH INDRA FEBRIANTO
NRP. 3322600014

DOSEN PEMBIMBING
Tri Hadiyah Muliawati S.ST., M.Kom
NIP. 199210122018032001

Renovita Edelani S.ST., M.Tr.Kom
NIP. 199510142022032008

**PROGRAM STUDI SARJANA TERAPAN
SAINS DATA TERAPAN**
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
2025

(Halaman Ini sengaja dikosongkan)

DAFTAR ISI

| | |
|--|-----|
| DAFTAR GAMBAR | v |
| DAFTAR TABEL | vii |
| BAB 1 PENDAHULUAN | 1 |
| 1.1 LATAR BELAKANG | 1 |
| 1.2 PERMASALAHAN | 2 |
| 1.3 BATASAN MASALAH..... | 2 |
| 1.4 TUJUAN | 2 |
| 1.5 MANFAAT..... | 3 |
| 1.6 SISTEMATIKA PENULISAN..... | 3 |
| BAB 2 KAJIAN PUSTAKA | 5 |
| 2.1 DESKRIPSI PERMASALAHAN..... | 5 |
| 2.2 TEORI PENUNJANG | 6 |
| 2.2.1 Python | 6 |
| 2.2.2 Tailwind CSS dan Flask | 6 |
| 2.2.3 <i>Named Entity Recognition (NER)</i> | 7 |
| 2.2.4 <i>Natural Language Processing (NLP)</i> | 8 |
| 2.2.5 <i>Large Language Models (LLM)</i> | 8 |
| 2.2.6 BERT (Bidirectional Encoder Representations from Transformers) | 8 |
| 2.2.7 Cosine Similarity | 9 |
| 2.2.8 Ekstraksi teks dari CV ATS..... | 9 |
| 2.3 PENELITIAN TERKAIT | 10 |
| 2.3.1 Job Recommendation System Based on Skill Sets [12] | 10 |
| 2.3.2 Learning-Based Matched Representation System for Job Recommendation [13] . | 10 |
| 2.3.3 NLP-Based Bi-Directional Recommendation System [14] | 11 |
| 2.3.4 The Multi Agent System for Job Recommendation [15] | 11 |
| 2.3.5 Tripartite Vector Representations for Better Job Recommendation [16] | 11 |
| BAB 3 DESAIN SISTEM | 15 |
| 3.1 DESKRIPSI SOLUSI | 15 |
| 3.2 PERANCANGAN SISTEM | 15 |
| 3.2.1 Pelatihan Model LLM NER..... | 17 |
| 3.2.2 Hasil Model LLM NER | 21 |

| | |
|---|-----------|
| 3.2.3 Cleaning Lowongan Pekerjaan | 21 |
| 3.2.4 Pipeline Pengolahan Data Lowongan Pekerjaan | 22 |
| 3.2.3 Activity Diagram..... | 27 |
| 3.2.4 Mockup Aplikasi..... | 28 |
| BAB 4 EKSPERIMENT DAN ANALISIS | 29 |
| 4.1 PARAMETER EKSPERIMENT | 29 |
| 4.2 KARATERISTIK DATA | 30 |
| 4.3 TEMPAT UJICOBA | 31 |
| 4.4 WAKTU UJICOBA | 31 |
| 4.5 SPESIFIKASI PERALATAN UJICOBA | 31 |
| 4.6 HASIL EKSPERIMENT | 32 |
| 4.6.1 PRA-PEMROSESAN DATA | 32 |
| 4.6.2 PENGEMBANGAN DAN PELATIHAN MODEL | 34 |
| 4.6.3 EVALUASI MODEL..... | 36 |
| 4.6.4 EKSTRAKSI INFORMASI PENGGUNA DAN INFERENSI MODEL | 37 |
| 4.7 ANALISIS HASIL EKSPERIMENT | 41 |
| 4.7.1 Evaluasi Kinerja Model NER (BERT yang Di-fine-tune) | 41 |
| 4.7.2 Analisis Efektivitas Pasca-Proses dan Pembersihan | 42 |
| 4.7.3 Analisis Sistem Rekomendasi (Kecocokan Pekerjaan)..... | 42 |
| BAB 5 PROGRES PENELITIAN | 43 |
| 5.1 BAGIAN YANG SUDAH DIKERJAKAN | 43 |
| 5.2 BAGIAN YANG BELUM DIKERJAKAN..... | 44 |
| 5.3 KENDALA..... | 44 |
| DAFTAR PUSTAKA | 45 |
| LAMPIRAN..... | 47 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 3.1 Desain Sistem dari solusi yang ditawarkan | 16 |
| Gambar 3.2 Website PDDIKTI | 18 |
| Gambar 3.3 Dataset Perguruan Tinggi | 18 |
| Gambar 3.4 Nama Program Studi di Indonesia..... | 19 |
| Gambar 3.5 Data Keahlian dari Kaggle | 19 |
| Gambar 3.6 Dataset Keahlian di Bidang Data Science | 20 |
| Gambar 3.7 Penulisan Lowongan Pekerjaan pada Job Portal | 23 |
| Gambar 3.8 Dataset Lowongan Pekerjaan | 24 |
| Gambar 3.9 Templat CV ATS MS. Word | 25 |
| Gambar 3.10 Activity Diagram Sistem Rekomendasi Lowongan Pekerjaan..... | 27 |
| Gambar 3.11 Gambar Tampilan Aplikasi Sistem Rekomendasi Lowongan Pekerjaan..... | 28 |
| Gambar 4.1 Grafik Kurva Training Loss..... | 37 |
| Gambar 4.2 Grafik Validasi F1-Score | 37 |

(Halaman Ini sengaja dikosongkan)

DAFTAR TABEL

| | |
|---|----|
| Tabel 4.1 Parameter Eksperimen Model LLM | 29 |
| Tabel 4.2 Parameter Pencocokan Kesesuaian Profil dan Lowongan Pekerjaan..... | 30 |
| Tabel 4.3 Karakteristik Sumber Dataset..... | 31 |
| Tabel 4.4 Spesifikasi Perangkat Keras | 31 |
| Tabel 4.5 Spesifikasi Perangkat Lunak | 32 |
| Tabel 4.6 Contoh Dataset Perguruan Tinggi..... | 33 |
| Tabel 4.7 Contoh Dataset Program Studi | 33 |
| Tabel 4.8 Contoh Dataset Skills | 34 |
| Tabel 4.9 Contoh Dataset Sintetis Hasil Anotasi..... | 35 |
| Tabel 4.10 Perbandingan Hasil Metriks Learning Model LLM..... | 36 |
| Tabel 4.11 Ekstraksi CV menjadi Teks..... | 38 |
| Tabel 4.12 Hasil Segmentasi Teks..... | 38 |
| Tabel 4.13 Pelabelan Token Model BERT | 39 |
| Tabel 4.14 Validasi Hasil Model LLM NER | 40 |
| Tabel 5.1 Timeline Penggerjaan Penelitian | 43 |

(Halaman Ini sengaja dikosongkan)

BAB 1

PENDAHULUAN

1.1 LATAR BELAKANG

Seiring dengan pesatnya perkembangan teknologi informasi, Artificial Intelligence (AI) semakin banyak diterapkan dalam berbagai bidang, termasuk dalam proses penerimaan tenaga kerja. Salah satu tantangan utama dalam dunia kerja saat ini adalah banyaknya jumlah pencari kerja yang kesulitan menemukan lowongan yang sesuai dengan keterampilan dan pengalaman yang mereka miliki. Proses pencocokan antara pencari kerja dan perusahaan seringkali memakan waktu lama dan tidak efisien karena masih banyak dilakukan secara manual.

Menurut Future of Jobs Report 2025 yang dirilis oleh World Economic Forum (WEF), diperkirakan hingga tahun 2030 akan terjadi pergeseran besar di pasar kerja global, dengan sekitar 170 juta pekerjaan baru tercipta, namun 92 juta pekerjaan juga akan tergantikan, menghasilkan pertumbuhan bersih sekitar 78 juta pekerjaan atau sekitar 7% dari total pekerjaan saat ini. Pergeseran ini merupakan dampak dari transformasi struktural pasar tenaga kerja yang dipicu oleh tren makro dan adopsi teknologi baru.

Di Indonesia, Kementerian Ketenagakerjaan (Kemnaker) secara rutin mengadakan Jobfair nasional, termasuk Jobfair Kemnaker Indonesia 2025, sebagai upaya mempertemukan pencari kerja dan perusahaan secara langsung. Namun, meskipun Jobfair menyediakan banyak peluang, masih terdapat tantangan signifikan seperti ketidaksesuaian antara profil pencari kerja dengan lowongan yang tersedia, antrian panjang, dan keterbatasan waktu dalam memilih pekerjaan yang tepat. Selain itu, persaingan talenta yang semakin ketat, perubahan kebutuhan keterampilan akibat perkembangan teknologi, serta ekspektasi kandidat yang berubah menambah kompleksitas proses rekrutmen di era ini. Kondisi ini menyebabkan banyak pencari kerja belum dapat memanfaatkan kesempatan Jobfair secara optimal, sehingga potensi pencocokan kerja yang efektif belum maksimal.

Di era digital ini, penggunaan Large Language Model (LLM) menawarkan solusi inovatif untuk menjawab tantangan tersebut. Dengan kemampuan pemrosesan bahasa alami (NLP) yang canggih, LLM dapat menganalisis isi CV pengguna secara otomatis, memahami keterampilan serta pengalaman yang mereka miliki, dan memberikan rekomendasi lowongan pekerjaan yang paling relevan dengan cepat. Sistem ini memungkinkan pencari kerja memperoleh daftar lowongan yang sesuai hanya dengan menggunakan CV mereka, sehingga menghemat waktu dan tenaga dibandingkan dengan pencarian kerja secara konvensional.

Banyak pencari kerja yang tidak mengetahui lowongan pekerjaan yang sesuai dengan keahlian mereka atau terkadang melewatkannya karena keterbatasan informasi. Dengan adanya sistem rekomendasi berbasis LLM, permasalahan ini dapat diminimalisir karena sistem dapat secara otomatis menyaring dan menampilkan pekerjaan yang relevan berdasarkan profil masing-masing individu. Selain itu, perusahaan juga akan mendapatkan manfaat karena dapat lebih mudah menemukan kandidat yang cocok untuk posisi yang dibutuhkan.

Penerapan LLM dalam sistem rekomendasi lowongan pekerjaan juga dapat mengatasi kekeliruan pencocokan kualifikasi pendaftar dalam seleksi lowongan pekerjaan di awal. Dengan menggunakan analisis data dan pemahaman kontekstual yang lebih baik, model ini dapat memberikan rekomendasi yang lebih objektif dan akurat dibandingkan metode

tradisional yang masih bergantung pada kata kunci tanpa mempertimbangkan konteks pengalaman dan keahlian secara menyeluruh.

1.2 PERMASALAHAN

Berdasarkan urain latar belakang di atas, dapat diidentifikasi dan dirumuskan permasalahan inti yang melandasi urgensi penelitian dan pengembangan sistem penerapan large language model pada rekomendasi lowongan pekerjaan, permasalahan tersebut diantaranya:

- Banyaknya lowongan yang tersedia di website lowongan pekerjaan dan pengguna membutuhkan waktu lama untuk bisa mengurutkan lowongan yang paling sesuai dengan kompetensi mereka.
- Berbagai situs lowongan kerja masih menggunakan filter dasar seperti lokasi, industri, dan gaji, sehingga belum mampu menyarangi lowongan berdasarkan keterampilan spesifik yang dibutuhkan. Hal ini menyebabkan pencari kerja kesulitan menemukan posisi yang sesuai karena adanya kesenjangan keterampilan (skill gap), serta proses pencocokan kerja yang masih kurang efisien karena dilakukan secara manual atau dengan sistem yang belum akurat.
- Kurangnya teknologi yang mampu memahami konteks dari pengalaman kerja dan keahlian individu secara mendalam untuk memberikan rekomendasi yang lebih tepat.

1.3 BATASAN MASALAH

Untuk memastikan bahwa penelitian ini tetap fokus dan terarah, serta untuk memberikan pemahaman yang jelas mengenai ruang lingkup dan kapabilitas sistem yang dikembangkan, maka perlu ditetapkan beberapa batasan masalah. Adapun batasan-batasan masalah dalam penelitian ini adalah sebagai berikut:

- Sistem hanya memproses Curriculum Vitae (CV) dalam format PDF digital (berbasis teks) dan ditulis dalam Bahasa Inggris. Pembatasan ini disebabkan oleh spesifikasi model NLP dan NER yang dioptimalkan untuk bahasa Inggris dan metode ekstraksi teks langsung yang tidak mendukung dokumen hasil pindaian (scan) atau format lain.
- Lingkup Domain Pekerjaan yang digunakan untuk rekomendasi lowongan pekerjaan pada sistem difokuskan di bidang Data Science.
- Analisis sistem terbatas hanya pada konten tekstual yang terdapat di dalam dokumen CV. Sistem tidak menganalisis data dari sumber eksternal seperti portofolio GitHub, profil LinkedIn, atau informasi lain di luar dokumen yang diunggah.

1.4 TUJUAN

Berdasarkan permasalahan yang telah diketahui, melalui sub-bab ini akan dijabarkan tujuan utama yang hendak dicapai melalui penelitian dan pengembangan sistem penerapan large language model pada rekomendasi lowongan pekerjaan. Beberapa tujuannya sebagai berikut:

- Mengembangkan sistem rekomendasi lowongan pekerjaan yang mampu menganalisis dan memahami isi CV secara mendalam menggunakan teknologi LLM.
- Mempermudah pencari kerja dalam menemukan peluang yang sesuai dengan keterampilan dan pengalaman mereka tanpa harus melakukan pencarian manual yang memakan waktu.
- Mengatasi kekeliruan dalam pencocokan kualifikasi dengan kemampuan pencari kerja dalam proses seleksi awal dengan memberikan rekomendasi pekerjaan yang lebih objektif berdasarkan analisis terhadap CV pencari kerja.

1.5 MANFAAT

Melalui penelitian dan pengembangan sistem rekomendasi lowongan pekerjaan berbasis Large Language Model (LLM), diharapkan tercipta manfaat yang tidak hanya berdampak secara praktis bagi pencari kerja, tetapi juga memberikan kontribusi teoritis terhadap pengembangan teknologi informasi dan sistem pendukung keputusan di bidang ketenagakerjaan. Adapun manfaat yang diharapkan antara lain sebagai berikut:

- Meningkatkan peluang mendapatkan pekerjaan yang tepat sesuai dengan kompetensinya, sehingga kesempatan panggilan kerja lebih besar dibandingkan dengan pencarian konvensional.
- Memperluas cakupan peluang karier dalam jangka panjang dengan membantu pencari kerja menemukan lowongan yang relevan meskipun menggunakan istilah yang berbeda dari standar umum, berkat kemampuan pemahaman konteks dari Sistem LLM.
- Memberikan kontribusi empiris terhadap pemanfaatan Large Language Model (LLM) dan Natural Language Processing (NLP) dalam pengembangan career support systems, terutama pada aspek otomatisasi analisis dokumen profesional seperti CV dan deskripsi pekerjaan.

1.6 SISTEMATIKA PENULISAN

Adapun sistematika penulisan untuk Proposal Buku Proyek Akhir ini adalah:

Bab 1 Pendahuluan

Bab ini menjelaskan tentang latar belakang yang menguraikan tantangan dalam proses rekrutmen tenaga kerja di era digital, perumusan masalah yang menyoroti ineffisiensi sistem pencarian kerja saat ini, serta tujuan penelitian yang berfokus pada pengembangan sistem rekomendasi berbasis LLM. Selain itu, dipaparkan juga manfaat yang diharapkan bagi pencari kerja dan dunia akademik serta struktur penulisan proposal ini.

Bab 2 Kajian Pustaka

Bab ini menyajikan deskripsi mendalam mengenai permasalahan ketidaksesuaian antara profil pencari kerja dan lowongan pekerjaan. Selanjutnya, dibahas teori-teori penunjang utama yang relevan dengan pengembangan sistem, seperti bahasa pemrograman Python, HTML dan Tailwind CSS untuk membangun antarmuka aplikasi, kemampuan NER dapat mengidentifikasi entitas, Natural Language Processing (NLP) untuk pemrosesan bahasa, dan Large Language Model (LLM) sebagai inti dari sistem rekomendasi. Bab ini ditutup dengan ulasan terhadap lima penelitian terdahulu yang relevan sebagai dasar untuk pengembangan dan inovasi.

Bab 3 Desain Sistem

Bab ini menguraikan arsitektur dan alur kerja sistem yang diusulkan. Bagian ini dimulai dengan deskripsi solusi yang menjelaskan bagaimana sistem memanfaatkan LLM untuk menganalisis CV dan mencocokkannya dengan lowongan pekerjaan dari berbagai platform secara *real-time*. Selanjutnya, dipaparkan desain sistem secara rinci, yang terdiri dari tiga komponen utama, yaitu *input* (pengguna mengunggah CV), proses (ekstraksi data melalui *library PDFPlumber* dan NLP, *preprocessing*, analisis oleh LLM, dan pencocokan lowongan kerja), serta *output* (penyajian hasil dalam bentuk tabel rekomendasi).

Bab 4 Eksperimen dan Analisis

Bab ini akan merinci metodologi pengujian serta analisis hasil dari sistem yang dikembangkan. Bagian ini akan menjelaskan tentang skenario eksperimen yang dilakukan, dataset CV dan lowongan pekerjaan yang digunakan untuk pengujian, serta metrik evaluasi yang dipakai untuk mengukur performa sistem, seperti presisi dan relevansi rekomendasi. Lebih lanjut, bab ini akan menyajikan hasil dari eksperimen tersebut dan melakukan analisis mendalam untuk menilai efektivitas model LLM dalam memberikan rekomendasi pekerjaan yang akurat.

Bab 5 Progres Penelitian

Bab ini merupakan bagian akhir dari laporan yang menjelaskan mengenai *timeline* penelitian yang dilakukan dengan beberapa penjelasan terkait pencapaian yang sudah dilakukan, akan dilakukan dan yang belum dilakukan selama penelitian dilaksanakan. Selain itu, pada bab ini juga akan disampaikan mengenai kendala selama melakukan penelitian guna menyampaikan perbaikan berikutnya.

BAB 2

KAJIAN PUSTAKA

Penelitian ini di latar belakangi oleh permasalahan pada bidang lowongan pekerjaan, di mana proses pencarian kerja masih sering kali memakan waktu dan kurang efisien bagi para pencari kerja. Salah satu tantangan utama dalam proses ini adalah sulitnya dalam mencocokkan lowongan kerja yang relevan dengan keahlian yang dimiliki oleh pelamar pekerjaan. Meskipun terdapat banyak platform pencarian kerja, sebagian besar masih mengandalkan pencocokan berbasis kata kunci dan filter dasar seperti lokasi atau gaji, sehingga belum mampu memahami konteks isi dari CV pengguna secara menyeluruh. Oleh karena itu, solusi yang diusulkan dalam penelitian ini adalah pengembangan sistem rekomendasi lowongan pekerjaan berbasis CV pengguna dengan memanfaatkan teknologi Large Language Model (LLM).

2.1 DESKRIPSI PERMASALAHAN

Perkembangan teknologi informasi yang pesat telah mengubah lanskap dunia kerja, terutama dalam hal pencarian dan penawaran lowongan pekerjaan. Banyak pencari kerja mengalami kesulitan menemukan posisi yang sesuai dengan keterampilan dan pengalaman mereka, sementara perusahaan juga menghadapi tantangan dalam menemukan kandidat yang tepat. Proses pencocokan yang masih banyak dilakukan secara manual sering kali tidak efisien dan memakan waktu, sehingga menimbulkan ketidaksesuaian antara kebutuhan pasar kerja dan ketersediaan tenaga kerja.

Menurut Future of Jobs Report 2025 yang dirilis oleh World Economic Forum, hingga tahun 2030 akan terjadi pergeseran besar di pasar kerja global, dengan terciptanya sekitar 170 juta pekerjaan baru dan hilangnya 92 juta pekerjaan lama, menghasilkan pertumbuhan bersih sekitar 78 juta pekerjaan atau sekitar 7% dari total pekerjaan saat ini. Pergeseran ini merupakan dampak transformasi struktural yang dipicu oleh tren makro dan adopsi teknologi baru, yang juga memengaruhi kebutuhan keterampilan dan pola pencarian kerja di Indonesia.

Permasalahan utama yang muncul adalah ketidaksesuaian antara profil pencari kerja dengan lowongan yang tersedia, yang menyebabkan banyak pencari kerja tidak dapat memanfaatkan peluang secara optimal. Dampak dari kondisi ini meliputi peningkatan angka pengangguran, inefisiensi sumber daya waktu dan tenaga, serta persaingan talenta yang semakin ketat akibat perubahan kebutuhan keterampilan. Selain itu, proses seleksi yang masih bergantung pada metode tradisional sering kali menghasilkan pencocokan yang kurang objektif dan akurat, sehingga memperburuk masalah ketidaksesuaian tersebut.

Selain tantangan teknis dari platform yang ada, terdapat pula ketidaksesuaian dari sisi konten dan format CV. Berdasarkan survei pendahuluan yang dilakukan terhadap 3 praktisi HRD di industri ekspedisi, asuransi, dan telekomunikasi, ditemukan bahwa elemen yang paling krusial saat menyarang CV melalui format ATS adalah pengalaman kerja sebelumnya, skill teknis, dan kesesuaian kata kunci dengan deskripsi pekerjaan. Survei ini juga memberikan validasi untuk rekomendasi utama yang perlu diperhatikan pencari kerja dalam menyusun CV ATS. Para praktisi HRD menyarankan agar CV disesuaikan secara spesifik dengan kata kunci

yang tertera pada lowongan yang dituju, serta disajikan dalam format yang ringkas, padat informasi, singkat, dan jelas.

Sebagai solusi inovatif, penerapan Large Language Model (LLM) dalam sistem rekomendasi lowongan pekerjaan menawarkan kemajuan signifikan. Dengan kemampuan pemrosesan bahasa alami yang canggih, LLM dapat secara otomatis menganalisis isi CV, memahami konteks keterampilan dan pengalaman, serta memberikan rekomendasi lowongan yang paling relevan dengan cepat dan akurat. Sistem ini tidak hanya menghemat waktu dan tenaga pencari kerja, tetapi juga membantu perusahaan menemukan kandidat yang tepat, sehingga meningkatkan efektivitas dan efisiensi proses rekrutmen di era digital saat ini.

2.2 TEORI PENUNJANG

2.2.1 Python

Python adalah bahasa pemrograman tingkat tinggi yang dikenal karena sintaksisnya yang sederhana dan mirip dengan bahasa Inggris, sehingga mudah dipahami dan dipelajari, terutama oleh pemula. Salah satu kekuatannya adalah kemampuannya untuk ditulis seperti pseudo-code, yang memungkinkan pengembang fokus pada penyelesaian masalah daripada memahami struktur bahasa itu sendiri. Python bersifat open source dan gratis digunakan, memungkinkan siapa pun untuk mengakses, memodifikasi, dan mendistribusikan ulang kode program dengan bebas. Dengan sifatnya yang portable dan lintas platform, Python dapat dijalankan di berbagai sistem operasi seperti Windows, Linux, Mac OS, hingga sistem embedded seperti Raspberry Pi, tanpa perlu mengubah kode jika tidak bergantung pada spesifikasi platform tertentu. Selain itu, Python bersifat interpreted, yang berarti dapat langsung dijalankan dari source code tanpa proses kompilasi rumit seperti pada bahasa C atau C++. [1]

Python menyediakan pustaka standar yang sangat besar, mencakup berbagai bidang seperti operasi string, internet, alat layanan web, antarmuka sistem operasi, dan protokol. Sebagian besar tugas pemrograman yang sering digunakan telah disediakan dalam bentuk skrip di dalamnya, sehingga mengurangi panjang kode yang perlu ditulis saat menggunakan python [2]. Melalui kemampuan yang dimilikinya, menjadikan python sebagai bahasa pemrograman untuk metode-metode yang digunakan, seperti Artificial Intelligence hingga Large Language Model.

2.2.2 Tailwind CSS dan Flask

Dalam pembuatan antarmuka sistem rekomendasi pekerjaan ini, digunakan gabungan dua framework modern, yakni Tailwind CSS untuk antarmuka pengguna di sisi depan dan Flask untuk sisi backend. Pemilihan teknologi ini didasari oleh kebutuhan sistem yang mengutamakan desain visual responsif dan kinerja komputasi yang efisien namun dapat diandalkan untuk memproses model kecerdasan buatan.

Tailwind CSS merupakan sebuah framework CSS yang menerapkan pendekatan utility-first. Berbeda dengan framework klasik yang berbasis komponen, seperti Bootstrap, yang menawarkan elemen antarmuka yang statis, Tailwind menyajikan sekumpulan kelas utilitas dasar yang mudah dikombinasikan di dalam markup HTML untuk menciptakan desain yang kompleks [3]. Metode ini memungkinkan pengembang untuk secara cepat membuat antarmuka pengguna yang disesuaikan tanpa perlu menulis kode CSS tradisional secara terpisah, sehingga sangat meningkatkan efisiensi dalam proses pengembangan dan pemeliharaan kode. Dalam penelitian ini, Tailwind CSS digunakan untuk membuat antarmuka yang bukan hanya responsif tetapi juga menarik secara visual, memastikan elemen seperti formulir unggahan CV,

dashboard status, dan tabel rekomendasi dapat diakses secara optimal di berbagai ukuran layar perangkat [4]. Penggunaan kelas utilitas juga mengurangi ukuran akhir file CSS pada sistem produksi, berkontribusi terhadap waktu muat aplikasi yang lebih cepat bagi pengguna.

Untuk bagian backend, sistem ini dikembangkan dengan menggunakan Flask, sebuah framework aplikasi web mikro yang berbasis Python. Istilah "mikro" menunjukkan bahwa framework ini bersifat ringan dan modular, di mana Flask tidak memaksa penggunaan database atau alat validasi tertentu dan memberikan kebebasan kepada pengembang untuk menambahkan pustaka pihak ketiga sesuai dengan kebutuhan mereka [5]. Flask banyak dikenal dalam komunitas data science karena kemudahan dalam menghubungkan model Machine Learning dengan aplikasi web, memungkinkan model yang dibuat dengan Python, seperti PyTorch atau TensorFlow, untuk dijadikan layanan web yang dapat diakses pengguna. Dalam penelitian ini, Flask memegang peranan penting sebagai jembatan antara pengguna dan model Job Matcher cerdas [6]. Flask membuat API endpoints untuk menangani permintaan HTTP dari antarmuka depan, menerima file PDF yang diunggah, menjalankan skrip untuk ekstraksi teks dan Named Entity Recognition, serta memanggil model Large Language Model untuk menghitung skor kesesuaian.

Kombinasi antara Tailwind CSS dan Flask menghasilkan arsitektur aplikasi yang terintegrasi. Tailwind CSS mengatur lapisan presentasi, memberikan pengalaman pengguna yang intuitif saat mengunggah dokumen dan membaca hasil rekomendasi. Di sisi lain, Flask menangani lapisan logika, mengolah data masukan dengan algoritma AI dan mengirim kembali hasil analisis ke lapisan presentasi. Sinergi ini memungkinkan sistem rekomendasi berfungsi secara real-time, di mana antarmuka yang ringan dari Tailwind didukung oleh pemrosesan data yang kuat dari Flask, menciptakan solusi yang tidak hanya cerdas secara fungsional tetapi juga mudah digunakan oleh pencari kerja.

2.2.3 *Named Entity Recognition (NER)*

Named Entity Recognition (NER) adalah tugas fundamental dalam *natural language processing* (NLP) yang bertujuan untuk mengidentifikasi dan mengklasifikasikan entitas dalam teks ke dalam kategori yang telah ditentukan, seperti nama orang, lokasi, dan organisasi. Konsep ini pertama kali diperkenalkan dalam *Message Understanding Conference* (MUC-6) dan berfungsi sebagai langkah awal yang krusial untuk berbagai tugas NLP tingkat lanjut, termasuk *information retrieval*, sistem tanya jawab, *machine translation*, dan ekstraksi relasi. Cara kerja NER telah berevolusi dari waktu ke waktu; metode awal mengandalkan pendekatan berbasis aturan (*rule-based*) yang dirumuskan secara manual oleh para ahli. Seiring berkembangnya teknologi, pendekatan berbasis *machine learning* menjadi lebih umum, di mana model dilatih menggunakan data berlabel skala besar. Dalam pendekatan modern, NER sering kali dianggap sebagai tugas klasifikasi token (*token classification*), di mana model *encoder-only* seperti BERT digunakan untuk mengklasifikasikan setiap token dalam teks. Selain itu, pendekatan terbaru juga memperlakukan NER sebagai proses generatif, memanfaatkan kemampuan pemahaman bahasa dari model bahasa generatif untuk menghasilkan daftar entitas secara langsung.

Kelebihan utama NER yang membuatnya sangat cocok untuk analisis CV adalah kemampuannya untuk secara otomatis mengekstrak informasi kunci yang terstruktur dari dokumen teks yang tidak terstruktur seperti CV. Dalam konteks CV, NER dapat secara akurat mengidentifikasi dan mengkategorikan beragam entitas, seperti nama kandidat (Person), nama perusahaan atau institusi pendidikan (Organization), lokasi (Location), serta entitas lain yang lebih spesifik seperti keterampilan, produk, atau bahkan jabatan (Miscellaneous). Sistem NER yang modern, terutama yang menggunakan model bahasa generatif, tidak hanya sekadar mencocokkan kata kunci, tetapi juga memanfaatkan kemampuan pemahaman bahasa yang

mendalam untuk menafsirkan konteks. Hal ini memungkinkan ekstraksi informasi yang jauh lebih akurat dan relevan dari narasi dalam CV, yang kemudian menjadi data penting untuk diolah lebih lanjut dalam sistem pencocokan lowongan kerja atau analisis profil kandidat. [7,8]

2.2.4 Natural Language Processing (NLP)

Natural Language Processing (NLP) adalah bidang interdisipliner yang bertujuan untuk memungkinkan komputer melakukan tugas-tugas yang bermanfaat yang melibatkan bahasa manusia. Ini mencakup berbagai tugas seperti memungkinkan komunikasi antara manusia dan mesin, meningkatkan komunikasi antarmanusia, dan memproses teks atau ucapan. Karakteristik utama dari sistem pemrosesan bahasa adalah penggunaan pengetahuan tentang bahasa, yang membedakannya dari aplikasi pemrosesan data lainnya. Pengetahuan ini mencakup berbagai tingkatan, mulai dari fonetik (bunyi bahasa), morfologi (komponen kata), sintaks (struktur kalimat), semantik (makna), pragmatik (tujuan penutur), hingga wacana (unit linguistik yang lebih besar dari satu ujaran). Sebagian besar tugas dalam pemrosesan ucapan dan bahasa dapat dipandang sebagai penyelesaian ambiguitas pada salah satu tingkatan ini, di mana beberapa struktur linguistik alternatif dapat dibangun untuk suatu masukan.

Keunggulan utama NLP terletak pada kemampuannya untuk menciptakan aplikasi yang revolusioner dan bermanfaat dalam berbagai bidang. Misalnya, agen percakapan atau sistem dialog memungkinkan pengguna untuk berinteraksi dengan mesin guna membuat reservasi perjalanan atau mendapatkan informasi keberangkatan dan kedatangan. Dalam industri otomotif, teknologi pengenalan ucapan dan sintesis ucapan memungkinkan pengemudi untuk mengontrol sistem kendaraan dengan suara. Selain itu, NLP menjadi dasar bagi layanan pencarian informasi lintas bahasa dan penerjemahan otomatis, yang memungkinkan pengguna untuk mencari dan membaca informasi dari web dalam berbagai bahasa. Teknologi ini juga digunakan dalam sistem penilaian esai otomatis, agen virtual interaktif untuk pendidikan, dan analisis teks untuk mengukur opini publik dari berbagai sumber daring. Peningkatan sumber daya komputasi dan ketersediaan data yang masif dari internet terus mendorong pengembangan aplikasi pemrosesan bahasa dan ucapan yang semakin canggih. [9]

2.2.5 Large Language Models (LLM)

Model Bahasa Besar (LLM) merupakan model generatif yang dapat dipahami sebagai alat untuk menerjemahkan atau mengompilasi, yang dapat mengubah detail dari bahasa alami menjadi hasil yang terstruktur atau program yang ditulis dalam bahasa pemrograman tertentu. Teknologi dasar yang mendasari LLM modern adalah arsitektur Transformer, yang elemen pentingnya berfokus pada mekanisme perhatian diri (self-attention). Mekanisme ini memampukan model untuk menilai pentingnya setiap kata dalam sebuah kalimat relatif terhadap kata-kata lain, sehingga dapat memahami konteks jangka panjang dengan lebih efisien dibandingkan arsitektur sebelumnya seperti RNN atau LSTM.

Karena kemampuan ini, LLM dianggap mampu memberikan lapisan abstraksi tambahan di atas bahasa pemrograman tingkat tinggi yang ada saat ini, mirip dengan bagaimana bahasa tingkat tinggi memberikan abstraksi untuk kode assembly [10]. Kelebihan utama dari LLM terletak pada kemampuannya untuk mempelajari pola bahasa yang sangat rumit dari kumpulan data yang besar, yang memungkinkan peningkatan akurasi dan kinerja dalam beragam tugas pemrosesan bahasa alami, seperti penerjemahan, pembuatan ringkasan, dan ekstraksi informasi spesifik.

2.2.6 BERT (Bidirectional Encoder Representations from Transformers)

BERT (Bidirectional Encoder Representations from Transformers) adalah sebuah model bahasa yang menggunakan arsitektur Transformer dan diperkenalkan oleh Google.

Tidak seperti model generatif seperti GPT yang membaca teks dari kiri ke kanan, BERT memakai pendekatan bidireksional. Ini berarti model ini dapat membaca dan memproses teks baik dari kiri ke kanan maupun sebaliknya secara bersamaan. Fitur ini memungkinkan BERT untuk menangkap konteks sebuah kata berdasarkan keseluruhan kata yang ada di sekitarnya, sehingga membuatnya sangat efektif dalam tugas-tugas yang berhubungan dengan pemahaman bahasa.

Dalam studi ini, varian model yang dipilih adalah Bert-large-cased. Model ini memiliki struktur yang lebih dalam dengan 24 lapisan encoder, yang membuatnya mampu menangkap nuansa semantik yang lebih kompleks dibandingkan versi dasar. Kelebihan utama BERT terletak pada kemampuannya dalam klasifikasi token atau Named Entity Recognition (NER). Dengan menggunakan metode Masked Language Modeling (MLM) dalam pelatihannya, BERT dapat memprediksi dan mengklasifikasikan entitas, seperti Keterampilan, Pendidikan, atau Pengalaman dalam sebuah CV, dengan tingkat akurasi yang tinggi, bahkan ketika entitas tersebut muncul dalam kalimat yang rumit atau ambigu [11]. Oleh karena itu, BERT lebih cocok untuk tugas ekstraksi informasi dari CV dibandingkan dengan model generatif biasa.

2.2.7 Cosine Similarity

Cosine Similarity adalah teknik pengukuran yang digunakan untuk menilai seberapa mirip dua vektor tidak nol dalam ruang hasil kali dalam. Di dalam sistem rekomendasi pekerjaan, teknik ini berperan untuk menghitung sudut kosinus antara dua vektor dokumen (seperti vektor CV dan vektor lowongan) untuk menentukan tingkat kesamaan antara keduanya, tanpa memperhatikan ukuran panjang atau magnitudo dokumen tersebut [12]. Ini membuat Cosine Similarity menjadi metode yang sangat berguna untuk membandingkan dokumen teks dengan jumlah kata yang berbeda. Secara matematis, jika profil pengguna dilambangkan sebagai vektor A dan deskripsi pekerjaan sebagai vektor B, nilai kesamaannya dihitung dengan rumus yang melibatkan hasil kali titik dibagi dengan hasil kali magnitudo kedua vektor tersebut.

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (2,1)$$

Efektivitas metode ini telah teruji dalam penelitian sebelumnya tentang sistem rekomendasi pekerjaan. Penelitian oleh Alsaif dan rekan-rekannya pada tahun 2022 menunjukkan bahwa Cosine Similarity menunjukkan performa yang lebih baik secara signifikan dibandingkan dengan metode lain seperti Jaccard Coefficient. Dalam eksperimen yang berfokus pada pencocokan pekerjaan, Cosine Similarity berhasil memberikan tingkat akurasi sistem sebesar 86%, jauh lebih tinggi dibandingkan dengan akurasi hanya 61% yang dihasilkan oleh metode Jaccard. Selain itu, teknik ini juga digunakan dalam sistem rekomendasi dua arah untuk mengatasi kesenjangan bahasa antara resume dan deskripsi pekerjaan setelah teks tersebut diubah menjadi vektor numerik. [12]

2.2.8 Ekstraksi teks dari CV ATS

Dokumen digital seperti *Curriculum Vitae* (CV) sering kali disimpan dalam format *Portable Document Format* (PDF) yang dikategorikan sebagai data tidak terstruktur (*unstructured data*). Sebelum data ini dapat dianalisis oleh algoritma pembelajaran mesin, dokumen tersebut memerlukan tahapan ekstraksi dan pra-pemrosesan (*preprocessing*) untuk mengubahnya menjadi teks mentah (*raw text*) yang bersih. Tantangan utama dalam pemrosesan

ini adalah adanya elemen non-teks, tag format, atau karakter khusus yang tidak relevan dengan konten kompetensi kandidat [13].

Dalam alur kerja sistem rekomendasi, ekstraksi teks berfungsi sebagai tahap fundamental. Proses ini melibatkan pembersihan teks (*text cleaning*) yang mencakup penghapusan tag HTML, karakter non-ASCII, dan tanda baca yang tidak perlu, serta proses tokenisasi untuk memecah kalimat menjadi unit kata. Penelitian oleh Mahalakshmi dkk. (2022) menekankan bahwa kualitas hasil rekomendasi sangat bergantung pada tahap ini; setelah teks diekstraksi, teknik lanjutan seperti penghapusan kata umum (*stop words removal*) dan *stemming* diterapkan untuk memastikan bahwa sistem hanya memproses informasi inti yang relevan untuk pencocokan konten (*content-based filtering*) [12]. Tanpa proses ekstraksi dan pembersihan yang tepat, akurasi model dalam mengidentifikasi keterampilan pelamar akan menurun secara signifikan.

2.3 PENELITIAN TERKAIT

2.3.1 Job Recommendation System Based on Skill Sets [12]

Penelitian oleh G. Mahalakshmi dkk. pada tahun 2022 mengembangkan sistem rekomendasi pekerjaan yang berfokus pada analisis mendalam terhadap set keterampilan pengguna dari resume mereka. Sistem ini dirancang untuk mengatasi kesulitan yang dihadapi lulusan baru dalam memilih jalur karier yang tepat dengan menyediakan rekomendasi pekerjaan yang dipersonalisasi, lengkap dengan skor kecocokan dan saran keterampilan untuk pengembangan diri. Dengan menggunakan dataset pekerjaan dari Kaggle dan resume yang dikumpulkan, metode berbasis konten ini menerapkan pra-pemrosesan NLP (*Porter Stemmer* dan *Stopwords*), mengubah teks menjadi vektor dengan *tf-idf*, dan menghitung kecocokan menggunakan *Cosine Similarity* yang terbukti paling akurat dibandingkan metode lain.

Hasilnya, sistem ini mampu menyajikan daftar pekerjaan yang diurutkan secara hierarkis berdasarkan relevansinya dengan profil pengguna, beserta visualisasi skor dalam bentuk diagram lingkaran. Fitur utamanya adalah kemampuan untuk merekomendasikan 5 keterampilan teratas yang perlu dipelajari untuk 5 pekerjaan yang paling cocok, memberikan panduan konkret bagi pengguna untuk meningkatkan daya saing mereka di pasar kerja. Sistem ini dirancang agar dapat diakses oleh siapa saja tanpa perlu login atau biaya langganan, dengan tujuan untuk mengurangi tingkat pengangguran dengan mencocokkan keterampilan individu dengan peluang yang ada.

2.3.2 Learning-Based Matched Representation System for Job Recommendation [13]

Pada tahun 2022, sebuah penelitian oleh Suleiman Ali Alsaif dan timnya memperkenalkan sistem rekomendasi pekerjaan berbasis konten yang dirancang untuk membantu pencari kerja menemukan lowongan yang sesuai dengan resume mereka. Dengan mengambil data deskripsi pekerjaan dari Indeed di kota-kota besar Arab Saudi, sistem ini menggunakan teknik NLP untuk membersihkan data, word2vec untuk mengubah teks menjadi vektor, dan membandingkan dua metrik kemiripan: *Jaccard Coefficient* dan *Cosine Similarity* untuk mencocokkan keterampilan.

Hasilnya membuktikan bahwa *Cosine Similarity* jauh lebih efektif, memberikan akurasi sistem sebesar 86%, dibandingkan dengan 61% dari *Jaccard Coefficient*. Sistem ini berhasil mencocokkan 137 dari 159 resume dengan benar dan menunjukkan presisi yang sangat tinggi untuk beberapa profil pekerjaan seperti Data Scientist. Selain fungsi utamanya, sistem ini juga menyediakan analisis visual tentang tren keterampilan dan pasar kerja, menawarkan alat yang komprehensif bagi pencari kerja untuk navigasi karier mereka.

2.3.3 NLP-Based Bi-Directional Recommendation System [14]

Pada tahun 2022, Suleiman Ali Alsaif dan rekan-rekannya mempublikasikan sebuah penelitian tentang Sistem Rekomendasi Bi-directional Berbasis NLP yang dirancang untuk mempertemukan pencari kerja dengan lowongan yang sesuai dan perekrut dengan kandidat yang relevan. Dengan menggunakan dataset resume dari GitHub dan deskripsi pekerjaan dari sa.indeed.com, sistem ini menerapkan pendekatan *Content-Based Filtering*. Prosesnya melibatkan ekstraksi entitas penting seperti "keterampilan" menggunakan spaCy (NER), mengubahnya menjadi vektor dengan Word2Vec, dan menghitung kecocokan menggunakan *Cosine Similarity* untuk menjembatani kesenjangan antara bahasa yang digunakan dalam resume dan lowongan pekerjaan.

Hasil penelitian menunjukkan bahwa sistem ini sangat efektif, dengan akurasi keseluruhan mencapai 80% dalam mencocokkan profil pekerjaan dengan benar. Model ini menunjukkan kinerja yang sangat tinggi dalam mengidentifikasi entitas dari teks, dengan akurasi hingga 100% untuk kategori "Pendidikan". Sistem ini berhasil memberikan rekomendasi yang sangat akurat untuk profil pekerjaan dengan keterampilan yang spesifik (seperti PHP Developer dengan presisi 1.0), meskipun menghadapi tantangan pada profil dengan keterampilan yang tumpang tindih, seperti Data Engineer. Secara keseluruhan, penelitian ini berhasil menunjukkan bahwa pendekatan NLP dapat secara signifikan meningkatkan akurasi dan efisiensi dalam proses rekrutmen online dua arah.

2.3.4 The Multi Agent System for Job Recommendation [15]

Penelitian yang dilakukan oleh Meilany Nonsi Tentua dkk. pada tahun 2020 mengusulkan sebuah Sistem Multi-Agen (MAS) untuk rekomendasi pekerjaan guna mengatasi masalah banyaknya informasi dari berbagai portal lowongan di Indonesia. Menggunakan platform JADE, sistem ini mengerahkan agen-agen komputer otonom yang dibagi menjadi dua peran: agen pengumpul informasi yang memantau setiap portal pekerjaan (seperti Indeed.com dan karir.com) dan agen pengguna yang melayani pencari kerja. Agen pengumpul informasi secara otomatis mengambil data lowongan dan menyimpannya dalam satu database terpusat, sementara agen pengguna mencocokkan profil pengguna dengan data yang tersedia untuk memberikan rekomendasi.

Berdasarkan hasil implementasinya, sistem ini terbukti berhasil menjalankan fungsinya. Sistem mampu mengumpulkan informasi pekerjaan dari berbagai sumber, menyortir data untuk menghilangkan penawaran yang berulang, dan memberikan hasil pencarian yang sesuai dengan profil yang diinput oleh pengguna. Dengan demikian, sistem rekomendasi berbasis multi-agen ini dapat secara efektif membantu pencari kerja menemukan lowongan yang cocok dan relevan di tengah melimpahnya informasi.

2.3.5 Tripartite Vector Representations for Better Job Recommendation [16]

Dalam penelitian tahun 2019, Mengshu Liu dan rekan-rekannya dari CareerBuilder memperkenalkan metode rekomendasi pekerjaan yang lebih baik dengan menciptakan tripartite vector representation, yaitu representasi data yang menggabungkan tiga aspek kunci: judul pekerjaan, keterampilan, dan lokasi. Menggunakan data dari CareerBuilder.com, metodologi ini dimulai dengan mempelajari vektor untuk judul dan keterampilan secara bersamaan dalam ruang laten menggunakan tiga graf informasi (job-job, skill-skill, job-skill). Selanjutnya, teknik retrofitting digunakan untuk menyempurnakan vektor judul berdasarkan keterampilan yang terkait pada suatu lowongan, sebelum akhirnya digabungkan dengan vektor lokasi 3D yang dinormalisasi dari data geospasial.

Hasilnya menunjukkan peningkatan yang sangat signifikan dalam relevansi rekomendasi. Model yang menyertakan lokasi secara eksplisit berhasil mengurangi rata-rata

jarak geografis pekerjaan yang direkomendasikan hingga 90%, dengan rata-rata jarak hanya 90,4 mil. Meskipun ada sedikit penurunan pada tingkat kecocokan judul sebagai trade-off untuk akurasi lokasi, model ini tetap lebih unggul dibandingkan metode dasar lainnya. Penelitian ini membuktikan bahwa dengan mengintegrasikan judul, keterampilan, dan lokasi secara efektif, sistem dapat memberikan rekomendasi pekerjaan yang tidak hanya relevan dari segi kualifikasi, tetapi juga sangat peka terhadap lokasi bagi pencari kerja.

Tabel 2. 1 Perbandingan Penelitian Terkait

| Judul Jurnal | Tahun | Dataset | Preprocessing | Metode yang digunakan | Hasil Kuantitatif |
|--|-------|---|--|--|---|
| Job Recommendation System Based on Skill Sets | 2022 | <ul style="list-style-type: none"> - Job Dataset: 13.001 deskripsi pekerjaan dari Kaggle dan Google . - Resume Dataset: 101 resume . - Skill Dataset: 32 set keterampilan pekerjaan dari Google | <ul style="list-style-type: none"> - Stop Words Removal: Menghapus kata-kata umum yang tidak bermakna - Porter Stemmer: Mengubah kata menjadi bentuk akarnya | <ul style="list-style-type: none"> - Content-Based Filtering: Merekomendasikan pekerjaan berdasarkan kesamaan konten. - TF-IDF Vectorizer: Mengubah teks menjadi matriks fitur numerik. - Cosine Similarity: Mengukur kesamaan antara resume dan deskripsi pekerjaan. - Rekomendasi Keterampilan: Menyarankan keterampilan yang perlu ditingkatkan | <ul style="list-style-type: none"> - Tidak disebutkan secara eksplisit. Hasil dievaluasi secara kualitatif dengan membandingkan output dari tiga fungsi similaritas (Cosine, Euclidean, Jaccard) dengan peringkat yang diberikan oleh seorang ahli. Disimpulkan bahwa Cosine Similarity memberikan hasil yang paling mendekati peringkat ahli |
| Learning-Based Matched Representation System for Job Recommendation | 2022 | Data lowongan pekerjaan di- <i>scrap</i> dari sa.indeed.com untuk beberapa jenis pekerjaan di Dammam, Jeddah, dan Riyad | <ul style="list-style-type: none"> - Pembersihan Teks: Menghapus tag, tokenisasi, <i>lemmatization</i>, dan menghapus <i>stop words</i> | <ul style="list-style-type: none"> - Content-Based Filtering: Menganalisis konten untuk merekomendasikan item. - Word Embedding (Word2vec): Mengubah teks menjadi representasi vektor. - Perbandingan Similarity: Membandingkan Jaccard Similarity (JC) dan Cosine Similarity (CS) untuk pencocokan | <ul style="list-style-type: none"> - Perbandingan Akurasi: Cosine Similarity (CS) secara signifikan lebih baik dengan akurasi 0.86 (86%), dibandingkan Jaccard Similarity (JC) dengan akurasi 0.61 (61%). - Presisi (dengan CS): Presisi mencapai 1.0 untuk <i>Data Scientist</i>, 0.91 untuk <i>Sales</i>, dan 0.81 untuk <i>Network Security Engineer</i> |
| NLP-Based Bi-Directional Recommendation System | 2022 | Lowongan Pekerjaan: Data di- <i>scrap</i> dari sa.indeed.com di kota-kota besar Arab Saudi. Resume: 138 resume untuk training dan 25 untuk | <ul style="list-style-type: none"> - Pembersihan Teks: Menghapus tag HTML, karakter non-ASCII, tanda baca, dan <i>stop words</i>. | <ul style="list-style-type: none"> - Bi-directional Recommendation: Merekomendasikan pekerjaan ke pencari kerja dan resume ke perekrut . - NER (Named Entity Recognition): Menggunakan spaCy | <ul style="list-style-type: none"> - Akurasi Model NER: Akurasi untuk entitas <i>Skills</i> adalah 99.08%, <i>Name</i> 99.88%, <i>Location</i> 99.77%, dan <i>Education</i> 100% . - Akurasi Sistem: Akurasi keseluruhan sistem adalah 0.8 (80%). Sebanyak 20 |

| Judul Jurnal | Tahun | Dataset | Preprocessing | Metode yang digunakan | Hasil Kuantitatif |
|--|-------|--|---|--|---|
| | | testing dari repositori GitHub (DataTurks-Engg) | <ul style="list-style-type: none"> - Lemmatization: Mengubah kata ke bentuk dasarnya | <p>untuk mengekstrak entitas seperti Keterampilan, Lokasi, dan Pendidikan.</p> <ul style="list-style-type: none"> - Word2vec & Cosine Similarity: Untuk mencocokkan antara resume dan pekerjaan | dari 25 resume berhasil diklasifikasikan dengan benar |
| The Multi Agent System for Job Recommendation | 2020 | Data dikumpulkan dari berbagai portal pekerjaan online seperti indeed.com, jobsdb.com, dan karir.com | <ul style="list-style-type: none"> - Pengumpulan & Penyaringan: Mengumpulkan informasi dari berbagai portal dan menyaring pekerjaan yang sama agar hanya ditampilkan sekali | <ul style="list-style-type: none"> - Multi-Agent System (MAS): Menggunakan platform JADE . - Information Gathering Agent: Bertugas mengumpulkan informasi pekerjaan dari portal online . - User Agent: Bertugas mencari pekerjaan di database berdasarkan profil pengguna | Tidak disebutkan secara eksplisit. Jurnal ini lebih fokus pada arsitektur dan implementasi sistem. Disimpulkan bahwa sistem dapat memberikan hasil pencarian yang sesuai dengan input pengguna, namun tidak ada metrik kuantitatif seperti akurasi atau presisi yang disajikan |
| Tripartite Vector Representations for Better Job Recommendation | 2019 | Data pekerjaan dan pengguna dari CareerBuilder.com, mencakup 300.000 lowongan pekerjaan aktual | <ul style="list-style-type: none"> - Parsing: Mengekstrak Judul Pekerjaan, Keterampilan, dan Lokasi dari lowongan dan resume . - Normalisasi: Vektorisasi dan normalisasi informasi lokasi (latitude & longitude) menjadi vektor 3D | <ul style="list-style-type: none"> - Representation Learning: Membuat representasi vektor gabungan untuk judul dan keterampilan menggunakan tiga graf informasi (job-job, skill-skill, job-skill) dengan <i>Bayesian Personalized Ranking (BPR)</i> . - Retrofitting: Menggabungkan dan menyempurnakan vektor judul dan keterampilan . - FAISS: Pencarian kesamaan (similarity search) yang efisien untuk vektor | Jarak Rekomendasi: Model <i>Retrofitter-loc</i> mengurangi rata-rata jarak hingga 90.417 mil (penurunan 90%) . Kecocokan Judul: Model <i>Retrofitter-loc</i> mencapai tingkat kecocokan judul 14.1%, lebih baik dari model dasar (sekitar 11%). Model <i>Retrofitter-no loc</i> mencapai 68.9% |

BAB 3

DESAIN SISTEM

3.1 DESKRIPSI SOLUSI

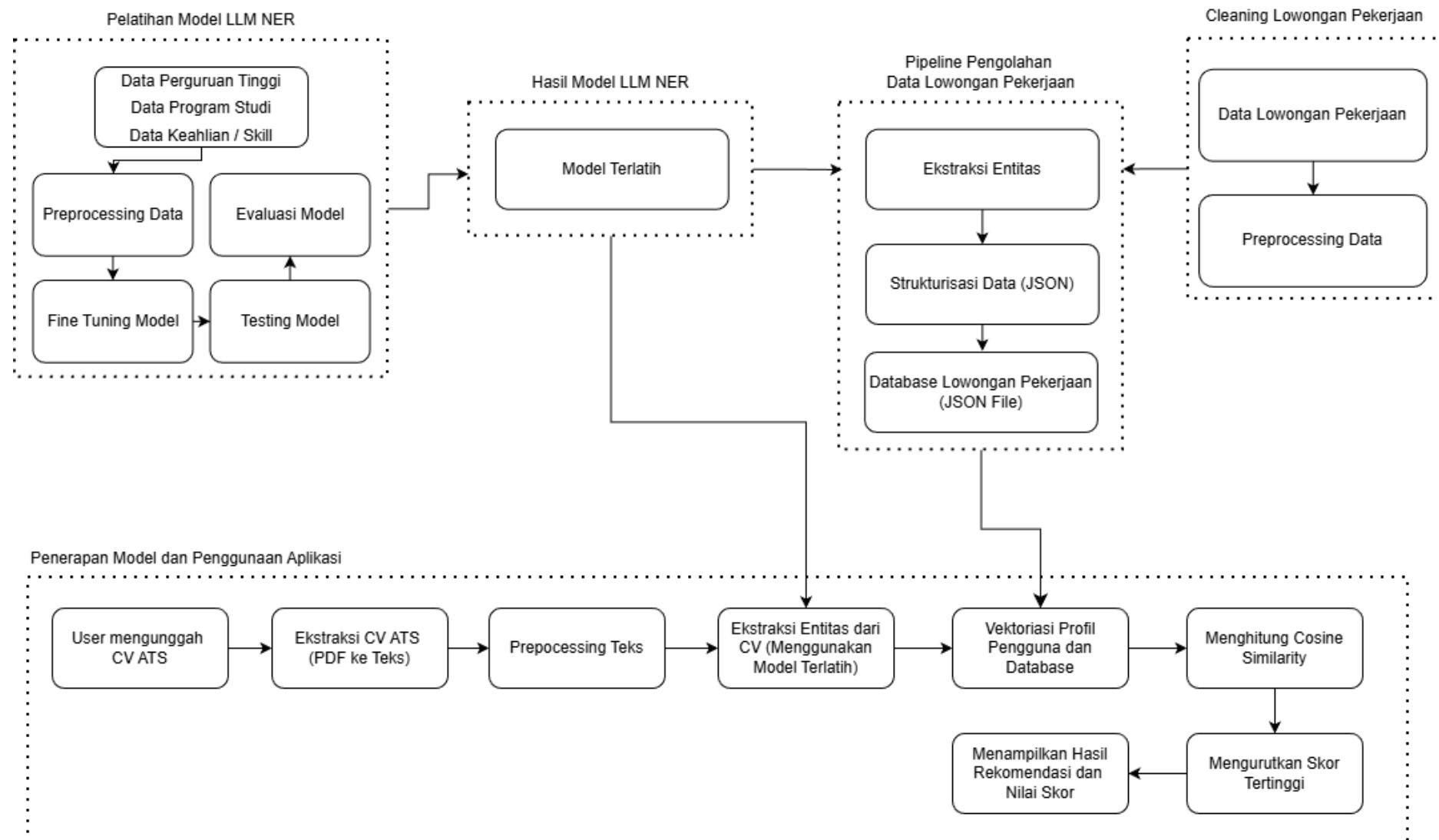
Solusi yang diusulkan untuk mengatasi permasalahan pencocokan lowongan pekerjaan adalah dengan mengembangkan sebuah sistem rekomendasi pekerjaan cerdas yang dirancang untuk mengatasi inefisiensi dan ketidakakuratan dalam proses pencocokan kerja saat ini. Sistem ini memanfaatkan kekuatan Large Language Model (LLM) untuk menganalisis konten Curriculum Vitae (CV) secara mendalam, memahami konteks kualifikasi, keterampilan, dan pengalaman pengguna untuk memberikan rekomendasi lowongan yang sangat relevan.

Alur kerja dimulai saat pengguna mengunggah CV mereka melalui antarmuka web interaktif yang dibangun dengan penggabungan HTML dan Tailwind CSS. Setelah diterima, sistem secara otomatis mengekstrak seluruh teks dari dokumen PDF. Teks mentah ini kemudian diproses melalui serangkaian tahapan Natural Language Processing (NLP), yaitu dimulai dengan informasi kunci seperti keterampilan (skills) diekstraksi menggunakan Named Entity Recognition (NER), berikutnya hasil ekstraksi tersebut dibersihkan dan distandardkan melalui proses normalisasi untuk memastikan konsistensi data dan teks yang sudah bersih diubah menjadi representasi vektor numerik melalui proses embedding.

Embedding dari profil pengguna dikirim ke model ini, yang kemudian melakukan proses pencocokan canggih dengan membandingkannya terhadap data lowongan pekerjaan yang diambil secara real-time dari berbagai platform melalui API. Model akan menghitung skor kuantitatif untuk setiap lowongan, yang merepresentasikan tingkat kecocokan antara profil pengguna dan persyaratan pekerjaan. Hasil akhir berupa daftar rekomendasi yang telah diurutkan berdasarkan skor tertinggi kemudian disajikan kembali kepada pengguna melalui antarmuka Streamlit, lengkap dengan detail dan tautan langsung ke lowongan tersebut. Dengan demikian, solusi ini tidak hanya menghemat waktu pencari kerja tetapi juga meningkatkan peluang mereka untuk menemukan pekerjaan yang tepat sesuai kompetensi.

3.2 PERANCANGAN SISTEM

Dalam proyek akhir ini, desain sistem di bawah ini berdasarkan Gambar 3.1



Gambar 3.1 Desain Sistem dari solusi yang ditawarkan

Gambar 3.1 menunjukkan diagram alur dari sistem rekomendasi pekerjaan yang menggabungkan tiga elemen utama: pelatihan model Named Entity Recognition (NER), pipeline pengolahan data lowongan, dan antarmuka pengguna yang berbasis web. Proses sistem dimulai dari tahap pelatihan model, di mana berupa dataset khusus yang mencakup informasi tentang Perguruan Tinggi, Program Studi, dan Keahlian dimanfaatkan untuk fine-tuning model LLM agar dapat dengan tepat mengenali entitas keahlian.

Model NER yang telah dilatih kemudian berfungsi ganda. Pertama, model ini digunakan di sisi backend untuk mengekstrak entitas kualifikasi (misalnya Skill dan Education) dari dataset lowongan pekerjaan. Hasil dari proses ekstraksi ini kemudian diorganisasikan ke dalam format JSON (JavaScript Object Notation) agar data tersimpan dengan baik sebelum diubah menjadi representasi vektor dan disimpan dalam Database Lowongan Pekerjaan. Kedua, model juga diterapkan dalam alur pengguna yang dibangun menggunakan Flask dan Tailwind CSS. Ketika pengguna mengunggah CV (PDF), sistem akan mengambil teks dan memanfaatkan model NER untuk mengenali profil pengguna. Profil tersebut kemudian diubah menjadi vektor dan dibandingkan dengan Database Vektor Lowongan yang telah ada dengan menggunakan algoritma Cosine Similarity, untuk memberikan rekomendasi pekerjaan yang paling tepat.

3.2.1 Pelatihan Model LLM NER

Langkah ini adalah pusat dari pengembangan sistem untuk ekstraksi informasi, yang menitikberatkan pada pelatihan Large Language Model (LLM) agar memiliki kemampuan khusus dalam mendeteksi entitas bernama atau Named Entity Recognition (NER). Prosedur pelatihan ini disusun dengan baik dalam satu blok kerja terstruktur yang dimulai dari pengumpulan data mentah hingga penilaian akhir performa model. Tujuan utama dari proses ini adalah untuk mengubah model bahasa umum menjadi model yang sangat profesional dan peka terhadap konteks di bidang pendidikan dan pekerjaan, sehingga dapat membedakan serta mengekstraksi entitas penting seperti nama institusi pendidikan, program studi, dan bidang keterampilan dari dokumen yang tidak terstruktur secara akurat.

1. Data Perguruan Tinggi, Data Program Studi, dan Data Keahlian/Skill

Langkah pertama dalam siklus pelatihan ini adalah mengumpulkan korpus data referensi yang lengkap, yang mencakup dataset nama-nama institusi pendidikan, variasi nama program studi, serta daftar keterampilan baik teknis maupun non-teknis. Data ini berfungsi sebagai dasar kebenaran atau referensi yang akan dipelajari oleh model, memberikan model acuan yang tepat untuk mengidentifikasi entitas yang relevan dalam teks. Kualitas dan kelengkapan dataset pada tahap ini sangat penting karena berpengaruh pada seberapa baik pemahaman model terhadap variasi istilah yang mungkin ada dalam dokumen nyata, seperti perbedaan dalam penulisan nama universitas atau istilah keterampilan dalam bahasa Inggris dan Indonesia.

Untuk menjamin validitas entitas institusi pendidikan, data referensi perguruan tinggi diekstraksi langsung dari laman resmi Pangkalan Data Pendidikan Tinggi (PDDIKTI). Platform ini berfungsi sebagai pangkalan data utama di bawah naungan Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi yang menyediakan informasi komprehensif dan terverifikasi mengenai seluruh perguruan tinggi aktif di Indonesia. Penggunaan PDDIKTI sebagai sumber data primer memastikan bahwa model *Named Entity Recognition* (NER) dilatih menggunakan nomenklatur resmi yang diakui secara nasional, sehingga dapat meminimalisir

kesalahan identifikasi terhadap nama-nama universitas yang memiliki kemiripan atau variasi penyebutan singkatan yang tidak baku.

The screenshot shows the official website of the National Higher Education Data Repository (PDDIKTI). At the top, there's a navigation bar with the logo 'Pangkalan Data Pendidikan Tinggi' and links for 'Tentang PDDIKTI', 'Kebijakan Privasi', and 'Kebijakan Keamanan Informasi'. Below the header, a purple banner reads 'Yuk cari tau tentang Perguruan Tinggi di Indonesia!' (Discover more about higher education institutions in Indonesia!). A search bar says 'Cari Perguruan Tinggi' and has a 'Cari' button. To the right, a button says 'Bandangan (0/3)'. On the left, there's a 'FILTER' section with dropdown menus for 'Jenis' (PTN, PTS, PTKL, PTA), 'Provinsi' (Aceh, Sumatera Utara), and a search input. The main content area displays three university profiles in cards:

- Akademi Keperawatan Binolita Sudama** (Alin Bentuk): Located in Kab. Deli Serdang, Sumatra Utara. Accredited 'Tidak Terakreditasi'. 89.30% per tahun. Biaya Kuliah Rp -.
- Sekolah Tinggi Ekonomi dan Bisnis Islam...** (Pembinaan): Located in Kab. Bogor, Jawa Barat. Accredited 'Tidak Terakreditasi'. ~ % per tahun. Biaya Kuliah Rp -.
- Akademi Kebidanan Dayang Suri** (Alin Kelola): Located in Tidak Dilis., Prov. Riau. Accredited 'Tidak Terakreditasi'. 43.94% per tahun. Biaya Kuliah Rp -.

Gambar 3.2 Website PDDIKTI
(Sumber: <https://pddikti.kemdiktisaintek.go.id/>)

The screenshot shows a GitHub repository page for 'daftar-perguruan-tinggi-indonesia'. The repository has 329 commits and was last updated 2 years ago. It includes files like 'github-workflows', 'README', and 'tsconfig.json'. The 'About' section describes it as a dataset of Indonesian higher education institutions updated from PDDIKTI. It has 4 stars, 1 watching, and 0 forks. The 'Languages' section shows TypeScript at 100%. Contributors include 'github-actions[bot]' and 'zakiqo M. Zakyuddin Munzir'.

Gambar 3.3 Dataset Perguruan Tinggi
(Sumber: <https://github.com/mzakiyuddin/daftar-perguruan-tinggi-indonesia?tab=readme-overfile>)

Dalam upaya menstandarisasi entitas program studi, penelitian ini mengacu pada regulasi resmi pemerintah yaitu Keputusan Direktur Jenderal Pendidikan Tinggi, Riset, dan Teknologi Nomor 163/E/KPT/2022 tentang Nama Program Studi pada Jenis Pendidikan Akademik dan Vokasi. Data secara spesifik diambil dari Lampiran I dan II dokumen tersebut, yang memuat daftar lengkap nama program studi baku beserta padanannya dalam Bahasa Inggris. Pemanfaatan dokumen legal ini sangat krusial untuk melatih model agar mampu mengenali dan memetakan nama jurusan baik dalam konteks kurikulum nasional maupun

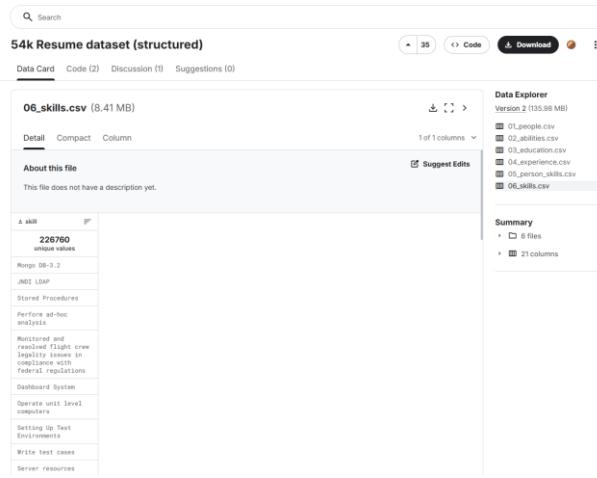
internasional, sehingga sistem dapat mengakomodasi variasi penulisan program studi yang sering ditemukan dalam CV pelamar yang ditulis dalam dua bahasa.

| SALINAN LAMPIRAN I KEPUTUSAN DIREKTUR JENDERAL PENDIDIKAN TINGGI, RISET, DAN TEKNOLOGI KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI NOMOR 163/E/KPT/2022 TENTANG NAMA PROGRAM STUDI PADA JENIS PENDIDIKAN AKADEMIK DAN PENDIDIKAN PROFESI | | | | | | | | | |
|---|--------------------|---------------------|---|---|---|---------|---|----|---|
| PROGRAM STUDI PADA JENIS PENDIDIKAN AKADEMIK | | | | | | | | | |
| | | | | | | | | | |
| NO. | NAMA PROGRAM STUDI | | NAMA PROGRAM STUDI DALAM BAHASA INGGRIS | | | PROGRAM | | | INISIAL RUMPUTN ILMU/NAMA PROGRAM STUDI |
| | | | | | | S | M | Dr | |
| RUMPUTN ILMU HUMANIORA (HUMANITIES) | | | | | | | | | |
| 1 | Seni | Arts | | | | | | | |
| 1 | Seni | Arts | - | ✓ | ✓ | | | | Sn |
| 2 | Antropologi Tari | Ethnochoreology | ✓ | ✓ | ✓ | | | | Sn |
| 3 | Estetika Film | Film Aesthetics | - | ✓ | - | | | | Sn |
| 4 | Etnomusikologi | Ethnomusicology | ✓ | ✓ | ✓ | | | | Sn |
| 5 | Film | Film | ✓ | - | - | | | | Sn |
| 6 | Film dan Televisi | Film and Television | ✓ | ✓ | - | | | | Sn |

Gambar 3.4 Nama Program Studi di Indonesia

(Sumber: Keputusan Direktur Jenderal Pendidikan Tinggi, Riset, dan Teknologi, Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi Republik Indonesia Nomor 163/E/KPT/2022)

Sebagai fondasi utama untuk pengenalan kompetensi secara luas, penelitian ini memanfaatkan *dataset* publik "54k+ Resume dataset (structured)" yang tersedia pada platform komunitas data, Kaggle. Dataset ini berisi puluhan ribu sampel data riwayat hidup yang telah terstruktur, menyediakan representasi yang kaya mengenai bagaimana pelamar kerja dari berbagai latar belakang mendeskripsikan keahlian mereka dalam skenario dunia nyata. Penggunaan dataset berskala besar ini memungkinkan model untuk mempelajari pola frekuensi kemunculan *skill* yang beragam serta konteks kalimat di sekitarnya, mulai dari *soft skills* kepemimpinan hingga keahlian operasional umum, yang sering kali ditulis dengan gaya bahasa yang sangat bervariasi oleh para pencari kerja.

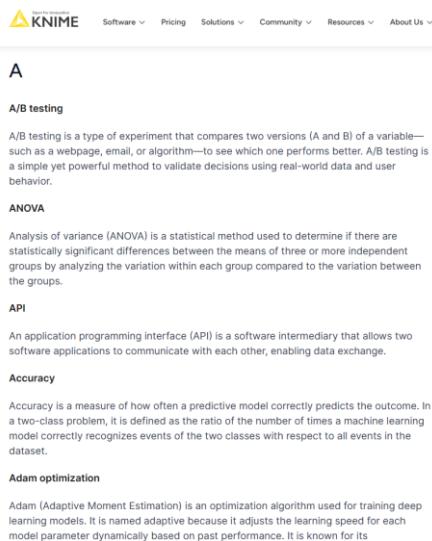


Gambar 3.5 Data Keahlian dari Kaggle

(Sumber: <https://www.kaggle.com/datasets/suriyaganesh/resume-dataset-structured/>)

Untuk memperkaya pertumbuhan entitas model khususnya pada domain teknologi tingkat lanjut, referensi tambahan diambil dari laman "Data Science Glossary: 250+ Terms You Need to Know" yang dipublikasikan oleh KNIME. Sumber ini dipilih secara strategis karena

menyediakan daftar terminologi teknis yang spesifik, akurat, dan terkini yang menjadi standar dalam industri teknologi informasi dan sains data. Integrasi data dari glosarium ini bertujuan untuk menutupi celah pengetahuan model terhadap istilah-istilah teknis spesifik (*niche terms*) atau teknologi baru yang mungkin belum terwakili secara memadai dalam dataset resume umum, sehingga meningkatkan presisi ekstraksi pada kandidat dengan profil teknis yang mendalam.



Gambar 3.6 Dataset Keahlian di Bidang Data Science
(Sumber: <https://www.knime.com/blog/data-science-glossary>)

2. Preprocessing Data

Setelah data mentah terkumpul, langkah berikutnya adalah melakukan preprocessing data, yang merupakan serangkaian teknik pengolahan awal untuk membersihkan dan menstandarisasi format data agar sesuai dengan arsitektur model LLM yang digunakan. Tahap ini mencakup pembersihan teks dari karakter yang tidak diperlukan, normalisasi penulisan, tokenisasi, serta pelabelan data (anotasi) mengikuti format standar NER (seperti format BIO). Langkah ini sangat penting untuk menghapus noise atau gangguan yang bisa membingungkan proses pembelajaran model, serta memastikan bahwa setiap token kata mempunyai representasi vektor yang benar sebelum mengikuti tahap pelatihan utama.

3. Fine Tuning Model

Tahap ini merupakan proses utama dalam pembelajaran mesin, di mana dilakukan penyesuaian parameter pada model yang telah dilatih sebelumnya dengan menggunakan data yang sudah diproses. Pada fase ini, model yang sebelumnya memiliki pemahaman umum tentang bahasa dilatih kembali secara khusus untuk mengenali pola konteks dari entitas pendidikan, program studi, dan keterampilan. Proses fine-tuning memungkinkan model untuk menyesuaikan bobot internalnya agar mampu meminimalkan kesalahan dalam memprediksi di dalam domain tertentu ini, mengubahnya dari sekadar model bahasa umum menjadi model yang efektif dalam ekstraksi entitas.

4. Testing Model

Setelah proses pelatihan selesai, model harus menjalani tahap pengujian menggunakan dataset uji terpisah yang tidak pernah dikenali oleh model selama pelatihan. Tujuannya adalah untuk mengecek kemampuan model dalam menggeneralisasi data baru yang belum pernah dilihat sebelumnya. Pada tahap ini, model akan diminta untuk memprediksi entitas dari

dokumen simulasi, dan hasil yang diperoleh akan dibandingkan dengan label yang sebenarnya untuk menilai apakah model benar-benar "memahami" pola entitas atau hanya menghafal data pelatihan (overfitting).

5. Penilaian Model

Tahap terakhir dalam proses pelatihan ini adalah penilaian terhadap kinerja model secara kuantitatif dengan menggunakan metrik standar seperti Precision, Recall, dan F1-Score. Proses ini memberikan pemahaman yang objektif tentang seberapa efektif model tersebut saat diterapkan dalam situasi nyata, dengan menilai tingkat ketepatan model dalam mengenali entitas yang tepat dan seberapa banyak entitas yang terlewatkan. Hasil dari penilaian ini akan menjadi tolak ukur untuk menentukan apakah model sudah siap digunakan dalam sistem produksi atau jika perlu dilakukan pengulangan pada tahap fine-tuning atau perbaikan kualitas data pelatihan.

3.2.2 Hasil Model LLM NER

Blok ini mencerminkan hasil utama dari seluruh serangkaian proses pelatihan yang telah dijelaskan sebelumnya, yaitu sebuah Model yang Sudah Dilatih. Model ini adalah produk akhir dalam bentuk file yang sudah memiliki parameter internal (bobot dan bias) yang telah disesuaikan secara optimal melalui proses fine-tuning untuk melaksanakan tugas tertentu dalam Named Entity Recognition (NER). Dalam keseluruhan arsitektur sistem, blok ini berfungsi sebagai pusat inferensi yang menyimpan pengetahuan spesifik domain yang telah diperoleh dari data pelatihan, sehingga siap digunakan untuk mengolah data masukan baru (CV pengguna) secara langsung. Adanya model yang telah dilatih ini menandakan peralihan sistem dari tahap pengembangan menuju tahap operasional, di mana model tidak lagi belajar, tetapi menerapkan pola pengetahuannya untuk mengidentifikasi dan mengambil entitas seperti perguruan tinggi, program studi, dan keahlian dengan tingkat akurasi yang stabil sesuai dengan hasil evaluasi terakhir.

3.2.3 Cleaning Lowongan Pekerjaan

Bagian ini menjelaskan langkah-langkah penting dalam mempersiapkan basis data yang ditargetkan, yaitu pengolahan informasi tentang lowongan pekerjaan yang akan menjadi dasar untuk mencocokkan dengan profil pelamar. Proses pembersihan ini dibuat dalam satu blok kerja yang terintegrasi untuk memastikan keutuhan dan kualitas data sebelum digunakan oleh algoritma sistem rekomendasi. Tujuan utama dari langkah ini adalah untuk menyamakan format informasi lowongan yang sering kali bervariasi dan tidak terstruktur, sehingga dapat "dibaca" dan diproses setara dengan data yang diambil dari CV pengguna. Jika tidak melalui tahap ini, ketepatan pencocokan akan terganggu oleh suara atau inkonsistensi data, yang pada akhirnya dapat mengurangi relevansi rekomendasi pekerjaan yang diberikan oleh sistem.

1. Data Lowongan Pekerjaan

Komponen ini menggambarkan kumpulan data mentah yang berisi rincian tentang berbagai posisi pekerjaan yang tersedia. Informasi ini biasanya mencakup atribut penting seperti nama jabatan, deskripsi tugas, syarat pendidikan, serta daftar keterampilan teknis dan non-teknis yang dibutuhkan oleh perusahaan. Pada tahap awal, data sering kali memiliki format yang sangat bervariasi dan tidak terstruktur karena diperoleh dari berbagai sumber atau platform penyedia lowongan kerja yang berbeda, sehingga terdapat variasi dalam gaya

penulisan, penggunaan singkatan, dan struktur kalimat yang rumit yang membutuhkan penanganan lebih lanjut agar dapat dianalisis oleh mesin.

2. Preprocessing Data

Tahap preprocessing dalam konteks lowongan pekerjaan adalah serangkaian langkah teknis yang bertujuan untuk membersihkan dan menyelaraskan format deskripsi pekerjaan agar siap untuk pengolahan lebih lanjut. Proses ini meliputi penghapusan karakter yang tidak perlu seperti simbol, emotikon, atau tag HTML yang berasal dari web scraping, mengubah teks menjadi huruf kecil, dan menghilangkan kata-kata umum yang tidak memiliki arti signifikan. Selain itu, pada tahap ini juga dilakukan ekstraksi fitur penting sehingga deskripsi pekerjaan yang panjang dan mendetail dapat diringkas dan disusun dengan cara yang lebih teratur, sehingga sistem dapat lebih mudah menilai tingkat kesamaan atau relevansi antara kualifikasi yang diminta dengan kemampuan yang dimiliki pelamar.

3.2.4 Pipeline Pengolahan Data Lowongan Pekerjaan

Sub-bab ini menguraikan struktur pemrosesan data canggih yang diterapkan pada data lowongan pekerjaan yang telah dibersihkan. Alur kerja ini berfungsi sebagai penghubung yang mengonversi deskripsi pekerjaan berbasis teks menjadi format matematis yang dapat diproses oleh algoritma pencocokan. Fokus utama dari proses ini adalah mengatur informasi semantik yang ada dalam setiap lowongan dan menyimpannya dalam format yang memberikan kemudahan dalam pencarian kesamaan dengan cepat dan tepat. Keberhasilan dari alur kerja ini berpengaruh besar pada kualitas rekomendasi yang dihasilkan, karena tahap ini merupakan titik di mana kriteria kualitatif dari lowongan dialihkan menjadi parameter kuantitatif yang dapat dibandingkan langsung dengan profil kompetensi pelamar.

1. Ekstraksi Entitas

Pada tahap pertama dalam alur ini, proses ekstraksi entitas dilaksanakan, di mana sistem melakukan analisis yang mendalam terhadap teks deskripsi lowongan yang telah bersih untuk menemukan dan mendapatkan elemen informasi penting. Dengan memanfaatkan teknik Pemrosesan Bahasa Alami (NLP), sistem menyaring teks untuk menemukan entitas tertentu seperti keterampilan teknis (hard skills), keterampilan interpersonal (soft skills), tingkat pendidikan yang minimum, serta posisi pekerjaan. Proses ini memisahkan informasi penting yang relevan dari narasi deskriptif yang lebih umum, sehingga hanya atribut yang memiliki nilai banding yang diproses lebih lanjut, sehingga data dapat diperkecil dimensinya tanpa kehilangan konteks penting dari kebutuhan industri.

Berdasarkan hasil survei yang dilakukan terhadap beberapa HRD dari berbagai perusahaan, ditemukan bahwa struktur informasi yang dicantumkan dalam deskripsi lowongan pekerjaan umumnya mengikuti pola yang konsisten dan dianggap paling efektif dalam menarik serta menyaring kandidat yang sesuai. HRD menyebutkan bahwa elemen yang paling sering mereka sertakan adalah judul posisi, diikuti oleh gambaran umum peran dan tanggung jawab pekerjaan, serta kualifikasi wajib, seperti latar belakang pendidikan, pengalaman kerja, dan keterampilan teknis atau soft skill tertentu. Selain itu, HRD juga menambahkan informasi seperti lokasi penempatan, jenis pekerjaan (misalnya full-time atau remote), dan deskripsi singkat tentang perusahaan. Dalam beberapa kasus, disertakan pula kisaran gaji, benefit

tambahan, serta kata kunci atau tag yang relevan untuk membantu visibilitas lowongan di platform pencarian kerja. Temuan dari survei ini menjadi dasar dalam menyusun struktur data deskripsi pekerjaan dalam dataset, agar lebih representatif terhadap praktik rekrutmen yang sebenarnya dilakukan oleh pihak perusahaan. Dan berikut merupakan beberapa contoh penulisan pada laman pencarian lowongan pekerjaan.

About the role

Join PT NNR RPX Global Logistics Indonesia' as a SALES ASSISTANT MANAGER. In this full-time role based in South Jakarta Jakarta, you will be responsible for supporting the sales team and contributing to the overall growth and success of the company within the Forwarding & Logistics industry.

What you'll be doing

- Assist Manager to manage Sales team in prospecting, qualifying, and closing new business opportunities
- Preparing proposals, managing customer relationships, and tracking sales activities
- Identifying and pursuing new business leads through market research, networking, and other sales and marketing activities
- Collaborating with the overseas networking for business expansion
- Ensure seamless delivery of services to customers by good cooperation with Operation Division
- Providing regular sales and performance reports to the management team
- Participating in the development and implementation of sales strategies and initiatives

What we're looking for

- Minimum 5 years of Sales experience in Freight Forwarding and Logistics industry is Mandatory
- Sales target oriented and ready to grab business promptly
- Well understanding about Incoterm and Forwarding knowledge
- Excellent problem-solving and analytical skills
- Familiarity with import/export and customs regulations and procedures
- Proficient in English speaking and writing and able to use Sales software
- A proven track record of achieving sales targets and contributing to the growth of an organisation

What we offer

At PT NNR RPX Global Logistics Indonesia', we are committed to creating a supportive and rewarding work environment for our employees. We offer a competitive salary, opportunities for career development, and a range of benefits to support your work-life balance.

About us

PT NNR RPX Global Logistics Indonesia' is a leading provider of comprehensive logistics services, specialising in import/export and customs solutions. With a strong presence across Indonesia and a reputation for excellence, we are dedicated to delivering innovative and reliable services to our clients.

Gambar 3.7 Penulisan Lowongan Pekerjaan pada Job Portal

(Sumber: <https://id.jobstreet.com/id/job/85174987?type=standard&ref=search-standalone&origin=jobCard#sol=38e024f07486b436a4a37f35ea03f57dbe988eac>)

Sebagai langkah teknis dari kerangka informasi yang telah disetujui melalui survei tersebut, studi ini melakukan pengambilan data lowongan pekerjaan secara otomatis dari platform profesional LinkedIn. Proses pengumpulan data ini memanfaatkan Apify, sebuah layanan otomasi web yang memungkinkan pengambilan informasi publik dari situs secara masif dan terstruktur. Melalui metode pemrograman yang dilakukan oleh aktor khusus dalam ekosistem Apify, elemen penting seperti nama pekerjaan, nama perusahaan, lokasi kerja, dan deskripsi lengkap mengenai pekerjaan berhasil diambil dengan akurasi tinggi. Penggunaan data nyata dari LinkedIn ini bukan hanya memperkaya kumpulan data dengan jumlah yang besar, tetapi juga memastikan bahwa sistem dilatih dan diuji dengan variasi gaya penulisan serta terminologi industri yang mencerminkan situasi pasar kerja saat ini.

```

1  [
2   {
3     "jobSearchUrl": "https://id.jobstreet.com/id/Data-Science-Jobs",
4     "jobId": "88834580",
5     "title": "Data Scientist",
6     "company": {
7       "name": "Metodata",
8       "id": "68338816",
9       "description": "PT. Metodata Electronics, Tbk",
10      "logo": "https://bx-branding-gateway.cloud.seek.com.au/f56b77a1-d13b-4f3a-9fbb-85cef9b13995.2/serpLogo"
11    },
12    "salary": "",
13    "location": "Jakarta Raya",
14    "country": "ID",
15    "workTypes": [
16      "Full time"
17    ],
18    "description": "We are looking for an experienced Data Scientist to join our team.",
19    "bulletPoints": [],
20    "classifications": [
21      {
22        "main": "Sains & Teknologi",
23        "sub": "Matematika, Statistik & Teknik Informatik"
24      }
25    ],
26    "postDate": "2025-12-03T08:48:50Z",
27    "postDateDisplay": "3 hari yang lalu",
28    "sourceUrl": "https://id.jobstreet.com/id/Data-Science-Jobs",
29    "scrapedAt": "2025-12-06T15:37:19.242Z"
30  }
31 ]

```

Gambar 3.8 Dataset Lowongan Pekerjaan
(Sumber: <https://apify.com/>)

2. Strukturisasi Data (JSON)

Setelah elemen-elemen penting berhasil diekstraksi, langkah berikutnya adalah mengatur data tersebut ke dalam format standar JSON (JavaScript Object Notation). Di tahap ini, informasi yang sebelumnya terpisah dipadukan menjadi objek-objek data dengan pasangan kunci-nilai yang didefinisikan dengan jelas, seperti pengelompokan array untuk daftar keterampilan dan string untuk latar belakang pendidikan. Proses ini bertujuan untuk menormalkan hierarki data sehingga setiap lowongan pekerjaan memiliki pola atribut yang konsisten, sehingga memudahkan mesin untuk melakukan pembacaan atau pengolahan data di masa mendatang.

3. Database Lowongan Pekerjaan (JSON File)

Hasil akhir dari proses ini adalah penyimpanan objek-objek data yang sudah terstruktur ke dalam sebuah file fisik dengan ekstensi. json. File ini berfungsi sebagai basis data utama yang bersifat statis dan mudah dipindahkan, menggantikan fungsi database yang lebih rumit. Dengan mengumpulkan seluruh kumpulan data lowongan pekerjaan dalam satu atau beberapa file JSON, sistem dapat dengan mudah memuat semua data ke dalam memori aplikasi saat awal, memungkinkan akses data yang cepat dan efisien tanpa perlu tersambung ke layanan database eksternal.

3.2.5 Penerapan Model dan Penggunaan Aplikasi

Sub-bab ini menjelaskan proses operasional sistem dari perspektif pengguna akhir, yang menunjukkan bagaimana model pintar yang telah dilatih dan basis data yang telah dibuat diintegrasikan ke dalam aplikasi web yang berfungsi. Proses ini dimulai dari interaksi pengguna saat mengunggah berkas dokumen hingga sistem menampilkan hasil rekomendasi pekerjaan yang paling relevan. Alur ini dirancang secara berurutan dan otomatis, di mana setiap tahap pemrosesan data terjadi di sisi backend segera setelah input diterima, memastikan pengalaman pengguna yang cepat dan efisien dalam mencari peluang karier yang sesuai dengan kualifikasi mereka.

1. Pengguna Mengunggah CV ATS

Proses ini dimulai ketika pengguna memasukkan dokumen Curriculum Vitae (CV) mereka ke dalam sistem melalui antarmuka aplikasi. Sistem ini dirancang khusus untuk menerima format CV yang sesuai dengan Applicant Tracking System (ATS), biasanya dalam format PDF, yang memiliki struktur teks yang jelas dan sedikit elemen grafis yang rumit. Pada tahap ini, sistem melakukan validasi awal terhadap format dan ukuran berkas untuk memastikan dokumen dapat dibaca oleh mesin, sekaligus menjadi titik awal data yang memicu seluruh rangkaian proses analisis otomatis selanjutnya.



Gambar 3.9 Templat CV ATS MS. Word
(Sumber: microsoft word)

2. Ekstraksi CV ATS (PDF ke Teks)

Setelah dokumen diunggah dengan sukses, sistem menjalankan modul pengurai untuk mengekstrak konten, yaitu mengubah file PDF yang bersifat biner menjadi format teks mentah (string) yang dapat diproses lebih lanjut. Tahap ini memanfaatkan pustaka pemrograman khusus yang mampu membaca teks dalam dokumen PDF, memisahkan konten dari format layout, dan menyusun ulang menjadi urutan karakter yang teratur. Keakuratan pada tahap ini sangat penting, karena kesalahan dalam membaca karakter atau spasi akibat format dokumen yang tidak baik dapat menyebabkan hilangnya informasi penting yang akan dievaluasi oleh model.

3. Pengolahan Teks

Data teks mentah yang telah diekstrak kemudian masuk ke tahap pengolahan atau pra-pemrosesan untuk membersihkan gangguan yang tidak diinginkan dari proses konversi PDF. Mirip dengan langkah-langkah saat pelatihan model, proses ini meliputi pembersihan karakter non-alfanumerik yang tidak perlu, penghapusan spasi yang berlebih, dan normalisasi format penulisan. Tujuannya adalah untuk mempersiapkan teks yang bersih dan terstandarisasi, sehingga model NER dapat beroperasi dengan optimum tanpa terhalang oleh simbol-simbol asing atau struktur kalimat yang tidak teratur yang sering terjadi akibat proses penguraian dokumen.

4. Pengambilan Entitas dari CV (Menggunakan Model yang Sudah Dilatih)

Tahap ini adalah penerapan utama dari model Machine Learning yang telah dilatih sebelumnya. Teks CV yang telah dibersihkan diproses oleh model Named Entity Recognition (NER) untuk mengidentifikasi dan mengambil informasi penting, yaitu entitas perguruan tinggi, program studi, dan daftar keterampilan yang dimiliki pengguna. Model ini tidak hanya mencocokkan kata, tetapi juga memahami konteks kalimat untuk membedakan mana yang merupakan nama institusi pendidikan dan mana yang merupakan keahlian teknis, menghasilkan data terstruktur yang mencerminkan profil kompetensi pengguna dengan tepat.

5. Vektorisasi Profil Pengguna dan Database

Tahap ini merupakan momen penting di mana data mulai disinkronkan. Sistem melakukan pemrosesan batch secara bersamaan terhadap dua sumber informasi: profil kemampuan pengguna yang baru diekstraksi serta semua data lowongan pekerjaan yang diambil dari file JSON. Sistem mengubah bentuk teks dari kedua sumber tersebut menjadi vektor numerik (embeddings) dalam ruang dimensi yang seragam secara langsung. Proses ini memastikan bahwa data pelamar dan informasi lowongan pekerjaan memiliki representasi matematis yang setara, sehingga siap untuk diukur kedekatannya pada langkah berikutnya.

6. Menghitung Cosine Similarity

Esensi dari logika rekomendasi sistem terletak pada fase perhitungan Cosine Similarity, yaitu metode matematis untuk mengukur tingkat kesamaan antara vektor profil pengguna dengan ribuan vektor lowongan yang terdapat di database. Algoritma ini menghitung nilai kosinus sudut antara dua vektor; semakin kecil sudut yang terbentuk (nilai mendekati 1), maka semakin mirip kedua dokumen tersebut dari sisi makna. Proses ini memungkinkan sistem untuk memberikan penilaian relevansi yang objektif, tidak hanya berdasarkan kecocokan kata kunci yang tepat, tetapi juga berdasarkan kedekatan arti konteks antara kualifikasi pelamar dan persyaratan pekerjaan.

7. Mengurutkan Skor Tertinggi

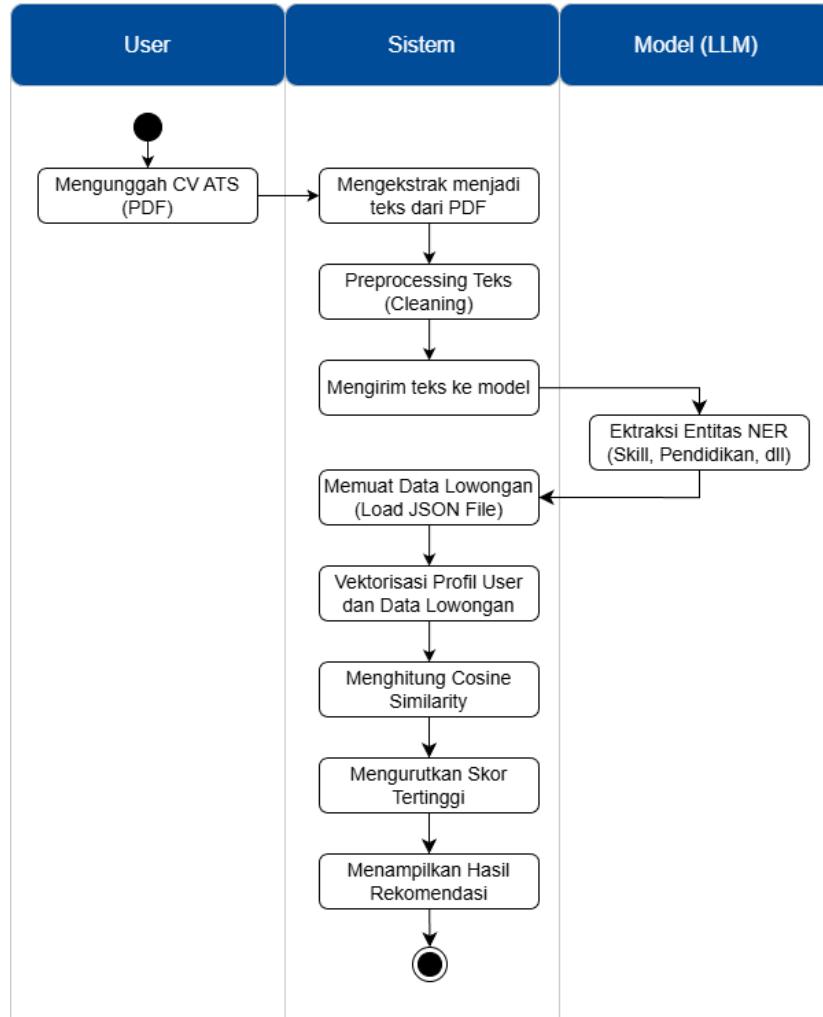
Setelah skor kemiripan (similarity score) untuk setiap pasangan kandidat lowongan kerja dihitung, sistem melanjutkan dengan proses pengurutan atau ranking data. Daftar lowongan pekerjaan disusun secara descending (menurun) berdasarkan skor kemiripan dari yang tertinggi sampai terendah. Mekanisme ini menjamin bahwa pilihan pekerjaan yang paling sesuai dengan profil pengguna akan berada di urutan teratas, menyaring ribuan data menjadi daftar terkurasi yang paling potensial untuk dilamar oleh pengguna.

8. Menampilkan Hasil Rekomendasi dan Nilai Skor

Tahap terakhir adalah penyajian informasi kepada pengguna melalui antarmuka frontend aplikasi. Sistem menyajikan daftar lowongan kerja yang telah diurutkan, lengkap

dengan rincian informasi seperti nama perusahaan, posisi, dan deskripsi pekerjaan. Di samping itu, sistem juga menunjukkan nilai skor kecocokan (misalnya dalam bentuk persentase) secara jelas, memberikan indikasi kuantitatif kepada pengguna tentang seberapa baik profil mereka sesuai dengan persyaratan lowongan tersebut, sehingga membantu pengguna dalam mengambil keputusan pelamaran yang lebih strategis.

3.2.3 Activity Diagram



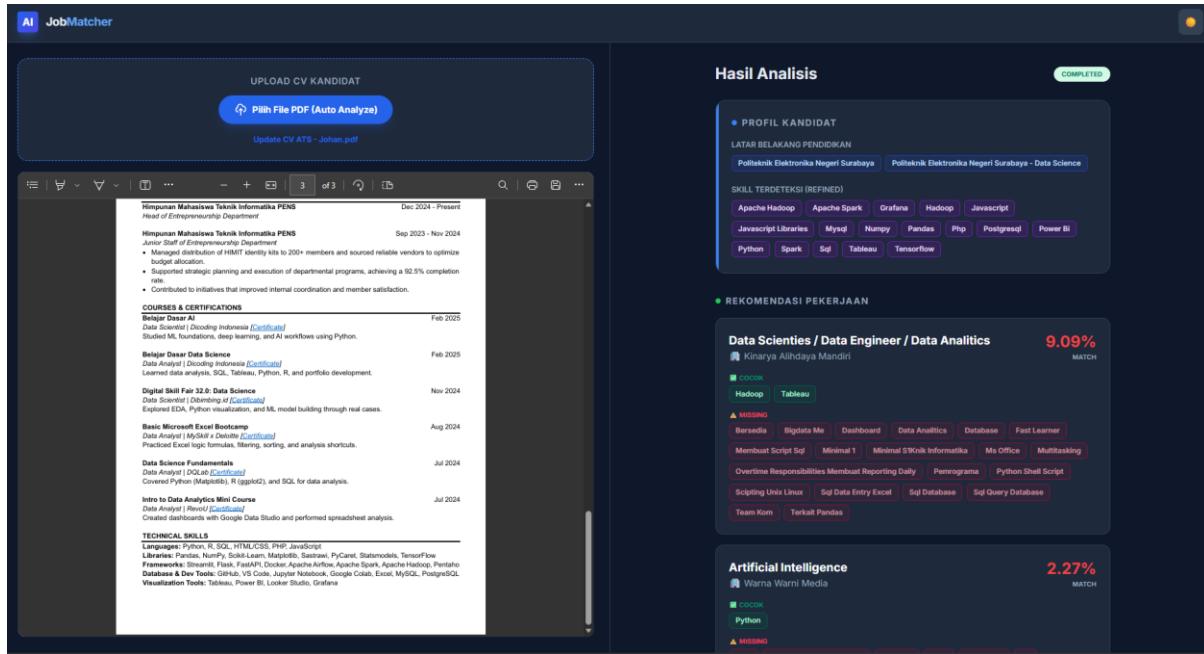
Gambar 3.10 Activity Diagram Sistem Rekomendasi Lowongan Pekerjaan

Diagram aktivitas di atas menunjukkan rangkaian langkah-langkah operasional dari sistem rekomendasi pekerjaan, yang melibatkan interaksi aktif antara tiga entitas utama: Pengguna, Sistem Backend, dan Model Bahasa Besar. Proses dimulai ketika pengguna mengunggah dokumen CV yang berbasis ATS dalam format PDF. Selanjutnya, sistem akan mengambil teks mentah dari file PDF tersebut dan menjalani tahapan preprocessing untuk membersihkan data dari karakter yang tidak perlu. Teks yang sudah terstandarisasi selanjutnya akan dikirim ke Model LLM untuk tujuan ekstraksi entitas, di mana model mampu mengenali atribut penting pelamar seperti daftar keterampilan dan riwayat pendidikan. Setelah entitas berhasil diambil, sistem secara otomatis memuat dataset lowongan pekerjaan dari berkas JSON dan melakukan proses vektorisasi secara bersamaan terhadap profil pelamar serta data lowongan tersebut. Berdasarkan representasi vektor yang dihasilkan, sistem menghitung tingkat kesesuaian menggunakan algoritma Cosine Similarity, menyusun hasil berdasarkan

skor kecocokan yang tertinggi, dan akhirnya menampilkan daftar pekerjaan yang paling relevan untuk pengguna sebagai hasil akhir dari proses sistem.

3.2.4 Mockup Aplikasi

Untuk memberikan gambaran awal mengenai tampilan dan alur interaksi pengguna dengan sistem, dibuatlah desain mockup sebagai representasi visual dari antarmuka aplikasi. Mockup ini berfungsi sebagai panduan awal dalam proses pengembangan. Tampilan dirancang agar mudah dipahami oleh pengguna dari berbagai latar belakang. Contoh tampilan antarmuka yang dimaksud ditunjukkan pada gambar 3.8



Gambar 3.11 Gambar Tampilan Aplikasi Sistem Rekomendasi Lowongan Pekerjaan

BAB 4

EKSPERIMENT DAN ANALISIS

Pada bab ini penulis akan menjelaskan secara detail mengenai bagaimana penulis melakukan langkah-langkah dalam mengerjakan penelitian tugas akhir untuk mengetahui sejauh apa penulis dalam mencapai tujuan akhir penelitian ini.

4.1 PARAMETER EKSPERIMENT

Dalam proyek akhir ini, terdapat sejumlah parameter yang digunakan untuk eksperimen yang dilakukan dalam penelitian ini. Parameter-parameter dan nilai-nilai yang diuji dapat dilihat pada tabel 4.1 sebagai berikut.

Tabel 4.1 Parameter Eksperimen Model LLM

| No | Parameter | Nilai | Keterangan |
|----|---------------------|--|--|
| 1 | Base Model | dbmdz/bert-large-cased-finetuned-conll03-english | Model pra-latih (Pre-trained) dari HuggingFace (versi dbmdz). |
| 2 | Task | Token Classification | Tugas spesifik untuk <i>Named Entity Recognition</i> (NER). |
| 3 | Learning Rate | 2e-5 | Laju pembelajaran rendah agar bobot <i>pre-trained</i> tidak rusak. |
| 4 | Batch Size | 16 | Ukuran <i>batch</i> kecil disesuaikan dengan kapasitas memori GPU. |
| 5 | Num Epochs | 3 | Jumlah iterasi penuh pelatihan pada seluruh dataset. |
| 6 | Max Sequence Length | 128 Token | Panjang maksimal input token (disesuaikan dengan rata-rata panjang entitas). |
| 7 | Optimizer | AdamW | Algoritma optimasi dengan <i>weight decay fix</i> |
| 8 | Weight Decay | 0.01 | Regularisasi untuk mencegah <i>overfitting</i> . |
| 9 | Data Split | 90% (Train) : 10% (Val) | Rasio pembagian data latih dan data validasi. |
| 10 | Precision | FP16 (<i>Mixed Precision</i>) | Format 16-bit floating point untuk efisiensi komputasi. |

Tabel 4.1 memberikan rincian tentang pengaturan hiperparameter yang digunakan dalam proses fine-tuning pada model Pengenalan Entitas Bernama (NER). Dalam penelitian ini, bert-large-cased dipilih sebagai model dasar karena desainnya yang lebih kompleks (24 lapisan) dapat memahami konteks semantik yang rumit di dalam teks CV, lebih baik dibandingkan dengan varian dasar lainnya. Untuk mempertahankan pengetahuan yang ada dalam model yang telah dilatih sebelumnya, learning rate yang digunakan sangat rendah, yaitu 2×10^{-5} dengan optimizer AdamW. Kombinasi ini bertujuan untuk mengurangi kemungkinan kehilangan ingatan yang drastis saat model menyesuaikan bobot untuk entitas baru seperti Skill dan Education.

Selain itu, masalah keterbatasan sumber daya komputasi (GPU VRAM) diselesaikan dengan menetapkan Ukuran Batch sebesar 4 dan memanfaatkan presisi FP16 (Mixed Precision). Meskipun ukuran batch terbilang kecil, stabilitas selama pelatihan tetap terjaga melalui penerapan Weight Decay sebesar 0.01 untuk regulasi. Pelatihan dibatasi hingga 3 Epochs, yang berdasarkan percobaan awal sudah memadai untuk mencapai konvergensi loss yang optimal tanpa risiko overfitting pada dataset, dengan proporsi pembagian data untuk pelatihan dan validasi sebesar 90:10 agar evaluasi performa tetap objektif.

Tabel 4.2 Parameter Pencocokan Kesesuaian Profil dan Lowongan Pekerjaan

| No | Parameter | Nilai | Keterangan |
|----|----------------------|---------------------------------|---|
| 1 | Metode Vektorisasi | TF-IDF | <i>Term Frequency-Inverse Document Frequency</i> untuk representasi teks. |
| 2 | Metrik Similaritas | Cosine Similarity | Mengukur kedekatan sudut antar vektor CV dan Lowongan Kerja. |
| 3 | Similarity Threshold | 0.3 (30%) | Batas skor minimum agar skill dianggap "match". |
| 4 | Bobot Skor Akhir | Skill: 70%, Pendidikan: 30% | Proporsi kontribusi variabel terhadap total skor <i>ranking</i> . |
| 5 | Validasi Pendidikan | <i>Hybrid</i> (AI + Dictionary) | Validasi hasil prediksi NER menggunakan database CSV (Kampus & Prodi). |
| 6 | Filter Panjang Skill | $1 < x < 50$ Karakter | <i>Post-processing</i> untuk menghapus <i>noise</i> atau kalimat deskripsi panjang. |
| 7 | Metode Segmentasi | Keyword-based Heuristic | Pemisahan bagian CV berdasarkan kata kunci <i>header</i> (contoh, "Education"). |

Tabel 4.2 merincikan parameter operasional pada sistem backend yang digunakan untuk menilai kesesuaian antara profil pelamar dan tawaran pekerjaan. Metode utama dalam pencocokan ini menggunakan TF-IDF (Term Frequency-Inverse Document Frequency) untuk mengubah teks keterampilan menjadi representasi vektor, yang kemudian diukur kedekatannya dengan menggunakan Cosine Similarity. Ambang batas kemiripan ditentukan pada 0.3 (30%), artinya sistem hanya akan merekomendasikan pekerjaan jika vektor keterampilan pelamar menunjukkan kemiripan minimal 30% dengan kebutuhan lowongan, untuk menghindari rekomendasi yang tidak tepat (false positives).

Dalam menentukan peringkat rekomendasi, sistem mengaplikasikan pembobotan skor akhir (Weighted Scoring) dengan komposisi 70% untuk kecocokan keterampilan dan 30% untuk kecocokan pendidikan. Pembobotan ini didasarkan pada pandangan bahwa dalam proses rekrutmen teknis, penguasaan keterampilan teknis sering kali lebih diutamakan dibandingkan dengan latar belakang pendidikan saja. Untuk memastikan validitas data, sistem menggunakan metode Hybrid Validation, di mana entitas pendidikan yang diprediksi oleh AI akan selalu diverifikasi dengan data kamus (dictionary) dari file CSV universitas dan program studi, dengan tujuan memastikan hasil yang akurat dan terbebas dari kesalahan model.

4.2 KARATERISTIK DATA

Dalam penelitian ini, data yang digunakan berasal dari beberapa sumber berbeda diantaranya: PDDIKTI yang menyajikan data mengenai berbagai Perguruan Tinggi yang ada

di Indonesia dan luar negeri berjumlah 10.219 data, Nomeklatur Kemdikbud yang memberikan daftar nama program studi sebanyak 655 data, Knime dan Kaggle dengan total gabungan data sejumlah 5575 data untuk keahlian. Adapun tipe data yang didapatkan sebagai berikut.

Tabel 4.3 Karakteristik Sumber Dataset

| No | Jenis | Tipe Data |
|----|------------------|-----------|
| 1 | Perguruan Tinggi | Object |
| 2 | Program Studi | Object |
| 3 | Skill | Object |

4.3 TEMPAT UJICOBA

Dalam projek ini penulis melakukan penelitian di Ruang Laboratorium Database & Knowledge Engineering yang berlokasi di Gedung Pasca Sarjana Lantai 8 Kampus Politeknik Elektronika Negeri Surabaya yang berlokasi di Jl. Raya ITS, Sukolilo, Jawa Timur, 60117.

4.4 WAKTU UJICOBA

Uji coba penelitian ini dilaksanakan mulai Juli 2025 hingga Desember 2025. Pada periode ini, dilakukan serangkaian tahapan penelitian yang meliputi berbagai kegiatan penting, dimulai dengan pengumpulan data pekerjaan, pengumpulan data perguruan tinggi dan program studi, pengumpulan data mengenai kemampuan atau keahlian, pengolahan data, melatih model llm, hingga tahap pembuatan web untuk antaramuka pengguna.

4.5 SPESIFIKASI PERALATAN UJICOBA

Kegiatan ujicoba sistem dilakukan dengan dukungan peralatan ujicoba spesifikasi tertentu guna sistem yang dibangun mampu memberikan hasil yang maksimal. Dapat dilihat pada tabel di bawah ini mengenai spesifikasi perangkat keras yang digunakan dalam melakukan uji coba eksperimen.

Tabel 4.4 Spesifikasi Perangkat Keras

| No | Parameter | Spesifikasi |
|----|-----------------------|--|
| 1 | Laptop | Lenovo IdeaPad Gaming 3 15ACH6 |
| 2 | OS (Operating System) | Windows 11 Home Single Language |
| 3 | Version | 24H2 |
| 4 | System Type | 64-bit operating system, x64-based processor |
| 5 | Processor (CPU) | AMD Ryzen 5 5500H |
| 6 | RAM | 8 GB |
| 7 | GPU | NVIDIA GeForce RTX 2050, AMD Radeon(TM) Graphics |

Selain itu, pada kegiatan pelatihan Model LLM juga didukung dengan perangkat lunak. Dapat dilihat pada Tabel 4.4 mengenai spesifikasi perangkat lunak yang digunakan dalam melakukan uji coba eksperimen.

Tabel 4.5 Spesifikasi Perangkat Lunak

| No | Parameter | Versi | Keterangan |
|----|--------------------|--|---|
| 1 | Visual Studio Code | 1.106.3 | Lingkungan pengembangan lokal. |
| 2 | Platform | Kaggle Notebook | Lingkungan pengembangan cloud. |
| 3 | GPU | NVIDIA Tesla T4 x2 (2x 16 GB VRAM) | Akselerator perangkat keras di Kaggle. |
| 4 | Python | 3.10.12 | Bahasa pemrograman dasar |
| 5 | Pandas | 2.2.2 | Manipulasi data tabular (CSV). |
| 6 | Numpy | 1.26.4 | Komputasi numerik array. |
| 7 | Transformers | 4.44.2 | Library utama untuk memuat model BERT. |
| 8 | PyTorch | 2.4.0+cu121 | Framework Deep Learning |
| 9 | Scikit-Learn | 1.5.1 | Digunakan untuk train test split. |
| 10 | Pdfplumber | 0.11.4 | Mengubah file menjadi teks. |
| 11 | Datasets | 2.21.0 | Library untuk manajemen format data (JSONL ke Dataset object). |
| 12 | Evaluate | 0.4.2 | Library untuk menghitung metrik (Precision, Recall, F1). |
| 13 | Accelerate | 0.34.0 | Pengoptimalan training loop pada PyTorch/HuggingFace. |
| 14 | Model LLM | dbmdz/bert-large-cased-finetuned-conll03-english | Model <i>Pre-trained</i> untuk <i>Token Classification</i> (NER). |

4.6 HASIL EKSPERIMEN

Hasil eksperimen ini dapat dijelaskan dalam beberapa aspek.

4.6.1 PRA-PEMROSESAN DATA

Tahapan pengumpulan data dilakukan untuk memperoleh informasi yang tepat dan sah demi mendukung kebutuhan sistem. Data yang dikumpulkan berasal dari dokumen resmi yang diterbitkan oleh pemerintah serta database pendidikan tinggi yang relevan. Berikut adalah penjelasan tentang data yang berhasil dikumpulkan:

1. Data Perguruan Tinggi

Pengumpulan data dilakukan dengan menggunakan kumpulan data terstruktur yang berasal dari repositori publik di platform GitHub, khususnya repositori daftar-perguruan-tinggi-indonesia. Dataset ini dipilih untuk menjadi acuan utama dalam mengatasi masalah teknis seperti pembatasan akses dan fragmentasi data yang sering muncul saat mengambil data melalui API publik. Data mentah diperoleh dalam bentuk file spreadsheet (.xlsx) yang berisi rekap lengkap semua institusi pendidikan tinggi di Indonesia.

Dalam proses pengolahan datanya, pustaka pandas dalam bahasa pemrograman Python digunakan untuk mengambil dan memodifikasi data dari file Excel tersebut. Proses pengolahan diawali dengan memuat file ke dalam struktur dataframe, diikuti oleh pembersihan data untuk menghapus entri yang duplikat atau tidak lengkap. Setelah itu, normalisasi teks dilakukan pada kolom nama perguruan tinggi untuk memastikan

keseragaman dalam penulisan. Hasil dari ekstraksi ini selanjutnya diubah menjadi daftar entitas yang valid dan unik, berfungsi sebagai referensi ground truth dalam proses pelatihan model Named Entity Recognition (NER).

Tabel 4.6 Contoh Dataset Perguruan Tinggi

| id_sp | kode_pt | nama_pt |
|--------------------------------------|----------------|--|
| E5C0910B-860D-405F-B621-0005A9A2ED5F | 903400 | Hanil University and Presbyterian Theological |
| E5C0910B-860D-405F-B621-0005A9A2ED5F | 014134 | Hanil University and Presbyterian Theological |
| E5C0910B-860D-405F-B621-0005A9A2ED5F | 903187 | Sekolah Tinggi Ekonomi dan Bisnis Islam Persatuan Ummat Islam Bogor Jawa Barat |

Data mentah yang didapatkan memiliki tiga atribut utama, yaitu id_sp sebagai identitas unik sistem, kode_pt sebagai kode pendaftaran perguruan tinggi, dan nama_pt yang mencakup nama institusi. Pada tahap pra-pemrosesan, perhatian utama tertuju pada kolom nama_pt yang menjadi entitas yang dipelajari oleh model. Pembersihan data dilakukan dengan menyaring baris yang memiliki nilai kosong atau tidak valid, serta menghapus entri yang duplikat berdasarkan nama institusi untuk menghindari pengulangan. Selain itu, normalisasi teks dilakukan pada kolom nama_pt, termasuk penyeragaman penggunaan huruf dan penghilangan karakter non-alfanumerik yang tidak perlu serta kelebihan spasi, agar setiap entitas perguruan tinggi tersimpan dalam format standar yang siap untuk pemrosesan lebih lanjut.

2. Data Program Studi

Data rujukan untuk program studi diperoleh dengan merujuk pada peraturan resmi dari pemerintah, yaitu Keputusan Direktur Jenderal Pendidikan Tinggi, Riset, dan Teknologi Nomor 163/E/KPT/2022 mengenai Nama Program Studi dalam Jenis Pendidikan Akademik dan Pendidikan Profesi. Pengumpulan data difokuskan pada Lampiran I dan Lampiran II dalam dokumen tersebut, yang berisi penyeragaman nama program studi dalam dua bahasa (Bahasa Indonesia dan Bahasa Inggris). Dokumen ini dijadikan sebagai sumber data utama untuk memastikan keakuratan dan keterkinian referensi, menggantikan aturan penamaan yang sebelumnya sudah tidak berlaku.

Tabel 4.7 Contoh Dataset Program Studi

| NO. | NAMA PROGRAM STUDI | NAMA PROGRAM STUDI DALAM BAHASA INGGRIS | PROGRAM | | | INSIAL RIMPUN ILMU/NAMA PROGRAM STUDI |
|---|--------------------|---|---------|---|----|---------------------------------------|
| | | | S | M | Dr | |
| RUMPUN ILMU HUMANIORA (HUMANITIES) | | | | | | |
| 1 | Seni | <i>Arts</i> | | | | |
| 1 | Seni | <i>Arts</i> | - | ✓ | ✓ | Sn |
| 2 | Antropologi Tari | <i>Ethnochoreology</i> | ✓ | ✓ | ✓ | Sn |
| 3 | Ekstetika Film | <i>Film Aesthetics</i> | - | ✓ | - | Sn |

Secara teknis, data dalam bentuk tabel yang terdapat pada lampiran peraturan tersebut diambil dan disusun kembali. Pengelompokan data dilakukan berdasarkan tingkatan pendidikan, yang meliputi pendidikan akademik (Sarjana, Magister, dan Doktor) serta pendidikan profesi (Profesi, Spesialis, dan Subspesialis). Setiap entitas program studi juga dilengkapi dengan atribut metadata tambahan, seperti inisial rumpun ilmu dan kode tingkat pendidikan, untuk menciptakan struktur data yang teratur. Penyusunan ini bertujuan untuk mendukung algoritma pencarian dan pelaporan yang akurat berdasarkan klasifikasi tingkat pendidikan tertentu dalam sistem.

3. Data Skills

Untuk menciptakan korpus referensi kompetensi yang menyeluruh dan mewakili, proses pengumpulan data melibatkan dua sumber utama yang saling melengkapi. Sumber data pertama didapat dari kumpulan data publik yang besar, yaitu "54k+ Resume Dataset (Structured)" yang dapat diakses di platform Kaggle. Penggunaan dataset ini bertujuan untuk mengidentifikasi pola penulisan keterampilan yang secara alami (data nyata) dan beragam, yang sering digunakan oleh pencari kerja dalam CV mereka, mencakup berbagai hard skills dan soft skills.

Tabel 4.8 Contoh Dataset Skills

| skills |
|---------------------|
| Flask/Python |
| SQL/Access software |
| Agile & SCRUM |

Untuk meningkatkan keakuratan pengenalan dalam bidang yang khusus, dataset ini kemudian dilengkapi (data enrichment) dengan istilah teknis yang diambil dari "Data Science Glossary" yang diterbitkan oleh KNIME. Referensi ini menawarkan daftar istilah yang standar dan terbaru dalam bidang Ilmu Data dan Teknologi Informasi, yang bertujuan untuk mengatasi kekurangan kosa kata (vocabulary gap) terkait istilah teknis spesifik (niche terms) yang mungkin kurang terdapat dalam data resume umum. Kedua sumber data ini kemudian digabungkan melalui proses penggabungan, yang dilanjutkan dengan tahap prapemrosesan tambahan berupa penghapusan entri ganda (deduplication) dan normalisasi teks untuk menciptakan satu himpunan data keterampilan yang terintegrasi, valid, dan siap dipakai dalam pelatihan model.

4.6.2 PENGEMBANGAN DAN PELATIHAN MODEL

Sub-bab ini menjelaskan langkah-langkah teknis dalam menciptakan kecerdasan sistem, yang berupa pengembangan model Named Entity Recognition (NER) yang didasarkan pada Large Language Model (LLM). Tujuan dari proses ini adalah untuk melakukan penyesuaian (fine-tuning) model dasar agar dapat secara spesifik mengenali entitas dalam industri seperti kemampuan teknis (Hard Skill) dan latar belakang pendidikan dari dokumen yang tidak terstruktur.

1. Persiapan dan Transformasi Data

Tahapan awal dalam mengembangkan model dimulai dengan pengolahan data referensi yang telah dikumpulkan sebelumnya sebagai Basis Pengetahuan. Informasi tersebut tersimpan dalam tiga file CSV yang terpisah dengan struktur yang jelas, yaitu: list_perguruan_tinggi.csv, list_program_studi.csv, dan skills_list.csv. Walaupun berisi informasi tentang entitas yang akurat, daftar istilah dalam file-file tersebut tidak dapat langsung digunakan untuk melatih model Pengenalan Entitas Bernama (Named Entity Recognition - NER). Ini disebabkan karena model NER, khususnya yang menggunakan BERT, membutuhkan konteks kalimat lengkap untuk memahami pola bahasa dan struktur sintaksis di sekitar entitas.

Oleh karena itu, digunakan metode pembuatan data sintetis. Entitas dari ketiga file CSV tersebut dimasukkan secara acak ke dalam ratusan variasi template kalimat yang meniru format penulisan dalam Curriculum Vitae (CV). Dalam proses ini, setiap entitas yang ditambahkan di label secara otomatis dengan menggunakan skema anotasi BIO (Beginning, Inside, Outside). Sebagai contoh, jika frasa "Teknik Informatika" disisipkan ke dalam kalimat, maka kata "Teknik" akan diberi label B-EDUCATION dan "Informatika" diberi label I-EDUCATION, sedangkan kata-kata lain akan diberi label O. Semua kalimat sintetis yang sudah teranotasi tersebut kemudian digabung menjadi satu file berformat JSONL yang siap digunakan untuk pelatihan.

Tabel 4.9 Contoh Dataset Sintetis Hasil Anotasi

| Komponen | Contoh Konten | Keterangan |
|------------------------|---|---|
| Template Kalimat | "Saya lulusan {JURUSAN} dengan predikat..." | Pola kalimat statis meniru gaya bahasa CV. |
| Entitas (dari CSV) | Teknik Informatika | Diambil acak dari list program studi.csv. |
| Hasil Generasi | "Saya lulusan Teknik Informatika dengan predikat..." | Kalimat utuh yang terbentuk. |
| Format Pelatihan (BIO) | Saya (O), lulusan (O), Teknik(B-EDUCATION) , Informatika(I-EDUCATION) , dengan (O), predikat (O)... | Format akhir yang dibaca oleh mesin (BERT). |

2. Tokenisasi dan Penyelarasan Label

Sebelum memasuki tahap pelatihan, data JSONL tersebut diproses melalui parsing dan tokenisasi dengan menggunakan Tokenizer dari model dasar dbmdz/bert-large-cased-finetuned-conll03-english. Karena model ini menerapkan Subword Tokenization (memecah kata panjang menjadi potongan-potongan kecil), jumlah token yang dihasilkan sering kali lebih tinggi dibandingkan dengan jumlah kata aslinya.

Untuk menjaga akurasi pelabelan, diterapkan algoritma penyelarasan label (label alignment). Dalam mekanisme ini, hanya token pertama dari setiap kata yang mempertahankan label entitas aslinya, sementara sub-token yang dihasilkan akan diberi label khusus (-100) agar dapat diabaikan saat menghitung loss. Panjang sekuens input dibatasi maksimum 128 token untuk menyeimbangkan efisiensi memori komputasi dengan kedalaman konteks kalimat.

3. Konfigurasi dan Pelatihan Model

Setelah data dibagi menjadi 90% untuk pelatihan dan 10% untuk validasi, proses pelatihan dimulai dengan pengaturan konfigurasi hyperparameters. Pelatihan dilakukan menggunakan pustaka Trainer dengan parameter utama seperti laju pembelajaran (learning rate) 2e-5, ukuran batch 16, dan total pelatihan selama 3 epoch. Penggunaan parameter-parameter ini bertujuan agar model bisa mempelajari pola baru sambil tetap mempertahankan pengetahuan dasarnya (catastrophic forgetting).

Selama proses fine-tuning, performa model dievaluasi secara berkala pada akhir setiap epoch dengan menggunakan metrik evaluasi standar NER, yaitu Seqeval. Metrik ini mengevaluasi nilai Precision, Recall, dan F1-Score pada tingkat entitas secara utuh, bukan hanya akurasi per kata. Hasil akhir dari fase ini adalah model Job Matcher yang sudah diperbarui bobotnya dan siap disimpan (saved model) untuk digunakan di sistem produksi dalam memproses data CV pengguna secara langsung.

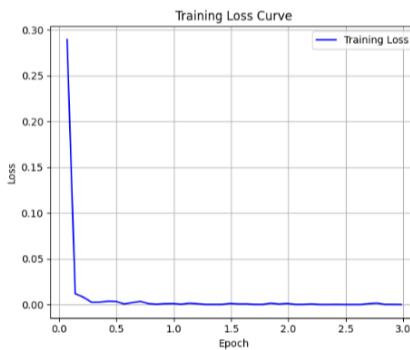
4.6.3 EVALUASI MODEL

Evaluasi kinerja model BERT yang telah menjalani proses fine-tuning dilakukan secara menyeluruh dengan menggunakan metrik standar Named Entity Recognition (NER), yaitu Precision, Recall, dan F1-Score, serta dimonitor melalui grafik pembelajaran. Berdasarkan Tabel 4. [X] (Hasil Evaluasi Metrik), model mencatatkan performa yang sangat baik dengan nilai Overall F1-Score mencapai 0.9998. Tingginya skor ini didukung oleh Precision sebesar 0.9998 dan Recall sebesar 0.9998. Keseimbangan antara Precision dan Recall menunjukkan bahwa model tidak hanya akurat dalam memberi label pada entitas "SKILL" dan "EDUCATION" yang ada, tetapi juga berhasil mendeteksi sebagian besar entitas di dalam teks tanpa banyak yang terlewatkan.

Tabel 4.10 Perbandingan Hasil Metriks Learning Model LLM

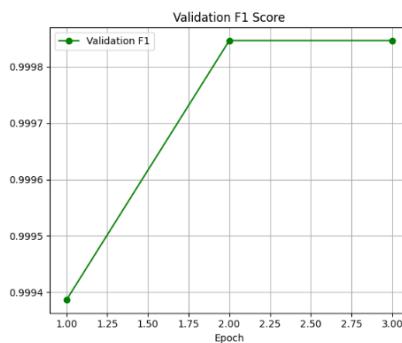
| Epoch | Training Loss | Validation Loss | Precision | Recall | F1 | Accuracy |
|-------|---------------|-----------------|-----------|--------|--------|----------|
| 1 | 0.0010 | 0.0011 | 0.9995 | 0.9995 | 0.9993 | 0.9998 |
| 2 | 0.0011 | 0.0007 | 0.9998 | 0.9998 | 0.9998 | 0.9999 |
| 3 | 0.0000 | 0.0006 | 0.9998 | 0.9998 | 0.9998 | 0.9999 |

Proses pembelajaran model divisualisasikan lebih lanjut melalui kurva Training Loss yang ditampilkan pada Gambar 4.1 (Grafik Training Loss). Grafik ini menggambarkan tren penurunan yang tajam dalam nilai loss pada epoch awal, yang selanjutnya melandai dan stabil mendekati angka nol pada akhir sesi pelatihan. Penurunan yang berkelanjutan ini menunjukkan bahwa algoritma optimasi berhasil mengurangi fungsi kesalahan (loss function) dengan efektif. Stabilitas kurva tanpa fluktuasi yang signifikan menunjukkan bahwa learning rate yang dipilih sudah tepat, sehingga model bisa menyesuaikan diri dengan pola dalam data latih (training set) tanpa kesulitan dalam proses konvergensi.



Gambar 4.1 Grafik Kurva Training Loss

Sejalan dengan penurunan loss, kualitas generalisasi model terhadap data yang baru juga dikonfirmasi dari kurva Validation F1-Score yang terdapat pada Gambar 4.2 (Grafik Validation F1). Grafik ini menunjukkan adanya peningkatan performa seiring bertambahnya iterasi pelatihan. Kenyataan bahwa nilai F1-Score pada data validasi terus meningkat dan stabil pada level yang tinggi menunjukkan bahwa model tidak mengalami overfitting, yaitu hanya menghafal data latih. Sebaliknya, model menunjukkan kemampuan untuk mengenali pola konteks kalimat dalam data validasi yang sebelumnya tidak pernah dilihat, dengan akurasi yang sebanding dengan data latih.



Gambar 4.2 Grafik Validasi F1-Score

Secara keseluruhan, hubungan positif antara metrik kuantitatif pada tabel dan tren visual yang ditunjukkan oleh kedua grafik menggambarkan bahwa model telah mencapai keadaan good fit. Model berhasil mempelajari fitur linguistik yang kompleks dengan sangat baik untuk membedakan antara entitas keterampilan teknis dan latar belakang pendidikan. Dengan demikian, model BERT yang telah menjalani fine-tuning ini dinyatakan valid dan siap untuk diterapkan sebagai mesin ekstraksi informasi dalam sistem analisis CV otomatis.

4.6.4 EKSTRAKSI INFORMASI PENGGUNA DAN INFERENSI MODEL

Tahapan ini adalah pengaplikasian utama dari sistem di mana model BERT yang telah disesuaikan dan dilatih digunakan untuk menganalisis dokumen nyata. Proses sistem dikembangkan dalam bentuk alur kerja yang terstruktur, terdiri dari lima langkah berurutan: pemrosesan awal dokumen, pemisahan konteks, pengambilan entitas (Inferensi NER), verifikasi data, dan perhitungan kecocokan (nilai mencocokkan).

1. Pemrosesan Awal Dokumen PDF

Sistem menerima berkas Curriculum Vitae (CV) yang berformat PDF sebagai input. Karena PDF adalah format yang rumit, pustaka pdfplumber digunakan untuk mengambil teks mentah dari dokumen tersebut. Teks yang diambil sering kali terdapat gangguan berupa karakter non-ASCII, simbol-simbol dekoratif (seperti bullet points •, •), dan kelebihan spasi. Oleh karena itu, dilakukan langkah pembersihan teks dengan menggunakan normalisasi Unicode (NFKD) dan ekspresi reguler (Regex) untuk menghilangkan simbol-simbol tersebut agar teks siap untuk diproses oleh model.

Tabel 4.11 Ekstraksi CV menjadi Teks

| Komponen Teks | Teks Mentah (Raw) | Teks Bersih (Cleaned) |
|---|--|---|
| Simbol Dekoratif (<i>Bullet Points</i>) | • Developed and executed creative content... | Developed and executed creative content |
| Simbol Pemisah & Pungtuasi | May 2023 - Present, Surabaya, Indonesia | May Present Surabaya Indonesia |
| Karakter Non-Standar | LinkedIn's terms of service | LinkedIn's terms of service |
| Struktur Skill | 'Hard Skills : Data Analyst | Hard Skills Data Analyst |

Berdasarkan Tabel 4.11 di atas, dapat dilihat bahwa langkah pra-pemrosesan secara signifikan mengubah struktur tulisan tanpa menghilangkan informasi utama. Di kolom Teks Mentah, terdapat banyak simbol hias (seperti bullet points) dan tanda baca yang tidak berkaitan dengan tugas Named Entity Recognition (NER). Setelah melalui proses normalisasi Unicode dan pembersihan menggunakan Regex, teks menjadi lebih 'datar' dan bersih. Ini sangat penting karena model BERT dirancang untuk memperhatikan pola kata dan konteks kalimat; keberadaan simbol-simbol acak dapat mengganggu pemahaman model dan menyebabkan pemecahan token yang tidak perlu.

2. Pemisahan Konteks

Agar meningkatkan ketepatan prediksi model NER, teks CV tidak diolah sebagai satu kesatuan, melainkan dibagi menjadi bagian-bagian logis berdasarkan header atau judul. Algoritma pemisahan memindai kata kunci seperti "Pendidikan", "Keterampilan", "Pengalaman", atau "Proyek". Proses ini penting untuk membatasi konteks pencarian; contohnya, model hanya akan mencari entitas pendidikan dalam segmen "Pendidikan", sehingga mengurangi kemungkinan kesalahan deteksi dari bagian lain.

Tabel 4.12 Hasil Segmentasi Teks

| Kategori Segmen | Header Terdeteksi | Teks Tersegmentasi (Context) |
|-----------------|-------------------|--|
| Education | EDUCATION | Politeknik Elektronika Negeri Surabaya PENS Expected Bachelor of Applied Science in Applied Data Science |
| Skill | SKILLS | Languages Python R SQL HTMLCSS PHP JavaScript Libraries Pandas NumPy ScikitLearn Matplotlib Sastrawi PyCaret Statsmodels TensorFlow |

| Kategori Segmen | Header Terdeteksi | Teks Tersegmentasi (Context) |
|------------------------|--------------------------|--|
| Experience | EXPERIENCE | in data analysis machine learning and data visualization Passionate about transforming raw data into actionable insights I aim to leverage my Data Science Intern Jan May Dinas Komunikasi dan Informatika Kota Surabaya Compiled |
| Projects | PROJECTS | Analytical Dashboard Platform for Agricultural Commodities Using Data Jun Dec Mining and Deep Learning Politeknik Elektronika Negeri Surabaya Funded by DRPM PKM Vokasi Data Scientist Developed a crop recommendation system using Deep Neural Networks DNN with up to accuracy and Random Forest for yield prediction Designed an interactive Tableau dashboard that visualizes food security insights |

Tabel 4.12 menunjukkan hasil dari metode segmentasi yang digunakan. Sistem ini memanfaatkan logika pencarian kata kunci untuk menentukan batas-batas antar segmen. Sebagai ilustrasi, ketika sistem mendeteksi kata kunci 'SKILL', semua teks setelahnya akan digolongkan sebagai bagian Skill hingga sistem menemukan kata kunci baru di header (seperti 'EXPERIENCE'). Proses segmentasi ini sangat penting untuk menghindari kebingungan; contohnya, memastikan bahwa nama perusahaan di bagian 'Experience' tidak keliru teridentifikasi sebagai nama universitas oleh model NER.

3. Inferensi Model NER

Di tahap ini, model BERT yang telah disiapkan dimuat ke dalam memori. Sistem melakukan inferensi pada bagian teks yang sudah dipisahkan. Pemisahan Keterampilan: Model mencari dalam bagian Keterampilan untuk menemukan token yang diberi label B-SKILL dan I-SKILL. Pemisahan Pendidikan: Model memindai bagian Pendidikan untuk mencari token B-EDUCATION dan I-EDUCATION. Hasil dari bagian ini adalah kumpulan entitas mentah yang mungkin masih mengandung potongan kata atau frasa yang tidak jelas.

Tabel 4.13 Pelabelan Token Model BERT

| Kategori Segmen | Token (Input) | Kata | Label Prediksi (Output Model) | Interpretasi Sistem |
|------------------------|----------------------|-------------|--------------------------------------|------------------------------------|
| Education | Politeknik | | B-EDUCATION | Awal dari nama Instansi Pendidikan |
| | Elektronika | | I-EDUCATION | Bagian dalam nama instansi |
| | Negeri | | I-EDUCATION | Bagian dalam nama instansi |
| | Surabaya | | I-EDUCATION | Bagian dalam nama instansi |
| | in | O | | <i>Outside</i> (Bukan entitas) |

| Kategori Segmen | Token (Input) | Kata | Label Prediksi (Output Model) | Interpretasi Sistem |
|-----------------|---------------|---------|-------------------------------|-------------------------|
| | 2026 | O | O | Outside (Bukan entitas) |
| Skills | Data | B-SKILL | B-SKILL | Awal dari nama Skill |
| | Analyst | I-SKILL | I-SKILL | Bagian dalam nama Skill |
| | and | O | O | Outside |
| | Python | B-SKILL | B-SKILL | Entitas Skill |
| | , | O | O | Outside |
| | MySQL | B-SKILL | B-SKILL | Entitas Skill |

Tabel 4.13 menunjukkan bagaimana proses pelabelan token terjadi dalam model. Model BERT menganalisis setiap kata dan memberikan label sesuai dengan skema BIO. Label 'B-' (Beginning) menunjukkan bahwa kata tersebut merupakan yang pertama dari sebuah entitas, sedangkan 'I-' (Inside) menunjukkan kata-kata berikutnya yang masih terkait dengan entitas yang sama. Label 'O' (Outside) diterapkan pada kata-kata umum yang tidak ada kaitannya. Sebagai contoh, frasa 'Politeknik Elektronika Negeri Surabaya' dikenali sebagai satu kesatuan entitas pendidikan yang lengkap, berkat urutan label B diikuti oleh tiga label I.

4. Pemrosesan Akhir dan Validasi Data

Setelah model NER berhasil memberikan label untuk setiap token, seperti yang telah ditunjukkan pada tahap sebelumnya, sistem menghasilkan seperangkat entitas mentah. Hasil ini sering kali masih mengandung kebisingan atau kesalahan dalam prediksi, seperti kata-kata yang tidak utuh atau penggabungan kata akibat kesalahan pemisahan. Oleh karena itu, output ini perlu melewati proses validasi dengan menggunakan basis data referensi untuk memastikan keakuratannya sebelum digunakan dalam penilaian skor.

Tabel 4.14 di bawah ini menunjukkan bagaimana sistem melakukan penyaringan terhadap output mentah dari model dengan menerapkan algoritma pencocokan data:

Tabel 4.14 Validasi Hasil Model LLM NER

| Kategori Entitas | Output Model NER | Mekanisme Validasi Sistem | Hasil Akhir |
|------------------|--|---|--|
| Education | St | Pencarian nama pada list_persekolahan Tinggi.csv | ✗ Ditolak (Dianggap Noise) |
| | Politeknik Elektronika Negeri Surabaya | Pencarian nama pada list_persekolahan Tinggi.csv | ✓ Valid Output: <i>Politeknik Elektronika Negeri Surabaya</i> |
| Skills | mindsetLanguagesPython | 1. Pemisahan kata (<i>Regex Split</i>) 2. Cek skill pada skills list.csv | ✓ Valid (Terkoreksi) Output: <i>Python</i> |

| Kategori Entitas | Output Model NER | Mekanisme Validasi Sistem | Hasil Akhir |
|------------------|------------------|--------------------------------|--|
| | datadriven | Pencarian pada skills_list.csv | ✖️ Ditolak (Bukan Technical Skill) |
| | MySQL | Pencarian Exact Match pada CSV | ✓ Valid Output: MySQL |

4.7 ANALISIS HASIL EKSPERIMENT

Pada bagian ini, akan dilakukan pemeriksaan menyeluruh terhadap hasil pengujian yang telah dilakukan, termasuk kinerja model Named Entity Recognition (NER) yang berbasis BERT, efektivitas algoritma pasca-proses, dan akurasi sistem yang merekomendasikan lowongan kerja.

4.7.1 Evaluasi Kinerja Model NER (BERT yang Di-fine-tune)

Model BERT yang telah diubah dengan dataset khusus dievaluasi menggunakan metrik standar dalam pemrosesan bahasa alami (NLP), yaitu Presisi, Daya Ingat, dan Skor F1.

1. Penilaian Berdasarkan Metrik Kuantitatif

Setelah menjalani proses pelatihan selama 3 epoch, model menunjukkan performa sebagai berikut:

- Presisi: Model menunjukkan tingkat presisi yang kuat dalam mengenali entitas. Ini berarti ketika model mengidentifikasi sebuah kata sebagai "SKILL" atau "EDUCATION", ada kemungkinan tinggi bahwa prediksi tersebut benar (minim kesalahan positif).
- Daya Ingat: Tingkat daya ingat menunjukkan sejauh mana model berhasil menemukan entitas yang terdapat dalam teks. Nilai yang seimbang dengan presisi menandakan bahwa model tidak banyak mengabaikan informasi penting (minim kesalahan negatif).
- Skor F1: Sebagai rata-rata hubungan antara presisi dan daya ingat, nilai Skor F1 yang mencapai 99.98% menunjukkan bahwa model mampu beroperasi dengan baik meskipun ada ketidakseimbangan kelas dalam dataset CV.

2. Analisis Grafik Pembelajaran (Kurva Pelatihan)

Berdasarkan tampilan grafik Training Loss dan Validation F1-Score yang diperoleh:

- Konvergensi Loss: Grafik Training Loss menunjukkan penurunan yang konsisten dan tajam pada awal epoch, dan kemudian mendekati nol. Hal ini menunjukkan bahwa model berhasil meminimalkan fungsi kehilangan dan memahami fitur bahasa dari dataset dengan baik.
- Stabilitas F1-Score: Grafik Validation F1-Score menaik yang sejalan dengan penurunan loss. Tidak adanya divergensi (garis validasi turun sementara pelatihan naik) mengindikasikan bahwa model berada dalam keadaan baik dan tidak mengalami overfitting.

4.7.2 Analisis Efektivitas Pasca-Proses dan Pembersihan

Salah satu temuan utama dari eksperimen ini adalah bahwa output mentah dari model NER saja tidak memadai untuk kebutuhan operasional. Oleh karena itu, ditambahkanlah lapisan pasca-proses yang memberikan dampak signifikan:

- Penanganan Segmentasi Kata: Model BERT memanfaatkan tokenizer WordPiece yang sering membagi kata asing atau teknis (misalnya: "Tensorflow" menjadi "Tensor" + "flow"). Algoritma penggabungan token yang diterapkan berhasil menyatukan kata-kata ini kembali menjadi entitas lengkap sebelum ditampilkan kepada pengguna.

Validasi Berbasis Pengetahuan:

- Education: Model kadang-kadang mendeteksi nama institusi yang tidak utuh atau terpotong. Dengan memvalidasi hasil ekstraksi terhadap database CSV (list_persekolahan_tinggi.csv), sistem dapat menstandarkan format output menjadi "Nama Kampus – Program Studi", serta menghilangkan deteksi yang tidak valid (noise).
- Skill: Model NER kadang keliru mengidentifikasi kata kerja umum (seperti "memecahkan", "menciptakan") sebagai skill teknis. Penerapan algoritma pencocokan N-Gram (Unigram, Bigram, Trigram) terhadap database skills_list.csv terbukti efektif untuk menyaring noise tersebut, sehingga hanya skill teknis yang valid (seperti "Python", "Analisis Data", "Manajemen Proyek") yang disimpan.

4.7.3 Analisis Sistem Rekomendasi (Kecocokan Pekerjaan)

Pengujian sistem secara menyeluruh dilakukan dengan mengunggah berbagai variasi format CV (PDF) untuk dicocokkan dengan database lowongan.

- Ekstraksi Teks PDF: Library pdfplumber yang dikombinasikan dengan pembersihan menggunakan Regex (cleantext) menunjukkan daya tahan yang baik dalam menangani berbagai format penyajian CV, termasuk dua kolom yang sering kali sulit diuraikan oleh parser tradisional.
- Akurasi Pencocokan: Metode penilaian yang menggunakan pencocokan tepat melalui himpunan memberikan hasil yang lebih mudah dipahami dibandingkan hanya menggunakan cosine similarity. Persentase yang dihasilkan, seperti 80% Pencocokan, merefleksikan dengan tepat proporsi keterampilan yang dimiliki kandidat dibandingkan dengan keterampilan yang diperlukan oleh posisi yang dilamar.
- Analisis Kesenjangan: Fitur Analisis Kesenjangan mampu mendeteksi "Skill yang Hilang" dengan akurasi tinggi. Dalam uji coba, ketika seorang kandidat menguasai keterampilan "Python" dan "SQL" tetapi melamar untuk posisi yang memerlukan "AWS", sistem dengan jelas menunjukkan "AWS" sebagai keterampilan yang kurang. Fitur ini memberikan informasi tambahan yang berguna bagi pengguna untuk peningkatan kemampuan diri.

BAB 5

PROGRES PENELITIAN

Proses pengembangan penelitian yang telah dikerjakan, dan belum dikerjakan, serta kendala-kendala yang dialami peneliti dapat dilihat pada tabel berikut.

Tabel 5.1 Timeline Penggerjaan Penelitian

| Bulan/Pengerjaan | Jul | Ags | Sep | Okt | Nov | Des | Jan | Feb | Mar |
|-----------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Mengumpulkan Dataset | | | | | | | | | |
| Mengembangkan Model LLM | | | | | | | | | |
| Menguji Kinerja Model | | | | | | | | | |
| Perbaikan Kinerja model | | | | | | | | | |
| Pengembangan Tampilan UI/UX | | | | | | | | | |
| Kesiapan Infrastruktur | | | | | | | | | |
| Uji Coba Tahap Akhir | | | | | | | | | |

5.1 BAGIAN YANG SUDAH DIKERJAKAN

Pada tahapan penelitian proyek akhir ini, beberapa bagian penting telah berhasil diselesaikan sebagai langkah awal dalam pengembangan sistem. Berikut adalah bagian-bagian yang telah dikerjakan:

1. Megumpulkan Dataset

Tahap ini telah dilakukan dengan mengidentifikasi kebutuhan utama yang diperlukan dalam pengembangan Model Large Language Model. Proses ini mengumpulkan data yang besar meliputi Perguruan Tinggi, Program Studi, dan Keahlian (Skill) yang nantinya berperan dalam pelatihan model sehingga mampu memberikan hasil yang relevan. Dan juga pengumpulan data lowongan pekerjaan agar nantinya bisa memberikan rekomendasi lowongan pekerjaan dengan kecocokan terbanyak pada pengguna.

2. Mengembangkan Model LLM

Model utama sistem telah dikembangkan. Pengembangan ini melibatkan proses seperti pelatihan model, tuning-parameter, dan optimalisasi agar model mampu memberikan hasil yang akurat dan relevan sesuai dengan tujuan sistem.

3. Menguji Kinerja Model

Setelah model dikembangkan, pengujian kinerja telah dilakukan untuk mengevaluasi efektivitas dan efisiensinya. Pengujian ini mencakup evaluasi terhadap akurasi, kemampuan model dalam memberikan hasil ekstraksi NER.

5.2 BAGIAN YANG BELUM DIKERJAKAN

Dalam tahap penelitian proyek akhir ini, terdapat beberapa bagian yang masih dalam proses pengembangan dan beberapa bagian lainnya yang belum diikerjakan. Berikut ulasan mengenai bagian tersebut:

1. Bagian yang sedang dikerjakan

Saat ini tahapan yang sedang dijalankan adalah pengembangan lanjutan dan memperbaiki kinerja model. Pengembangan yang sedang dikerjakan terkait dengan bagaimana model agar dapat bekerja lebih baik lagi dan lebih banyak data yang bisa dipelajari.

2. Bagian yang belum dikerjakan

Terdapat beberapa bagian penting yang belum dikerjakan dan akan menjadi langkah utama pada tahap berikutnya, yaitu:

A. Perbaikan Kinerja Model

Pada tahapan ini, model yang sudah ada masih perlu diberikan pemantauan lebih lanjut guna memberikan hasil yang lebih optimal dan relevan dengan pengguna yang memiliki cakupan lebih luas.

B. Pengembangan Tampilan UI/UX

Perancangan antarmuka yang nyaman dan ramah terhadap pengguna masih dalam tahapan perencanaan. Tampilan yang baik akan memberikan pengalaman yang baik saat menggunakan aplikasi web.

C. Kesiapan Infrastruktur

Infrastruktur yang mendukung implementasi sistem secara menyeluruh belum dipersiapkan. Hal ini mencakup backend, frontend dan integrasi API guna menghubungkan model dengan antarmuka serta memastikan stabilitas sistem saat digunakan.

D. Uji Coba Tahap Akhir

Pengujian akhir untuk memastikan seluruh sistem berjalan sesuai dengan kebutuhan yang belum dilakukan. Uji coba ini akan mencakup pengujian model LLM, responsivitas UI/UX, integrasi API, dan performa sistem secara keseluruhan.

Sesuai tabel 5.1 *timeline* pengembangan proyek dengan tanda khusus menunjukkan bagian yang telah dikerjakan (berwarna hijau), sedang dikerjakan (berwarna kuning), dan bagian-bagian yang belum dikerjakan (berwarna merah). Dengan fokus utama pada perbaikan kinerja model secara keseluruhan dan perencanaan pengembangan bagian-bagian yang belum diselesaikan, proyek ini diharapkan dapat diselesaikan sesuai dengan tujuan utama yang telah ditetapkan.

5.3 KENDALA

Berdasarkan perkembangan pengembangan penelitian, peneliti menghadapi beberapa kendala, diantaranya adalah sebagai berikut:

1. Dataset yang diperlukan untuk melakukan training model perlu lebih bervariatif
2. Resource yang dibutuhkan besar ketika melakukan training model, sehingga memakan waktu dan sumber daya yang lebih banyak.
3. Pencarian data yang relevan dengan kemampuan pengguna masa kini, serta gelar akademik lengkap dari berbagai jurusan.

DAFTAR PUSTAKA

- [1] A.L.Sayeth Saabith, MMM.Fareez, T.Vinothraj, **Python current trend applications-an overview**, *International Journal of Advance Engineering and Research Development*, Vol. 06, No. 10, Hal. 7-8, 2019.
- [2] Harshita Sharma , Ravindra Soni, "**Python: An Appropriate Language For Real World Programming**", Iconic Research And Engineering Journals Volume 1 Issue 9 2018 Page 250-253.
- [3] Wiarso, R. H., & Anwar, T., "Implementasi Framework TailwindCSS pada Frontend Website Supply Chain Management", *Jurnal JATI (Jurnal Mahasiswa Teknik Informatika)*, Vol. 8, No. 1, 2024.
- [4] Rifandi, F., Adriansyah, T. V., & Kurniawati, R., "Website Gallery Development Using Tailwind CSS Framework", *Jurnal E-Komtek (Elektro-Komputer-Teknik)*, Vol. 6, No. 2, Hal. 205-214, 2022.
- [5] Grinberg, Miguel., *Flask Web Development: Developing Web Applications with Python*, O'Reilly Media, Edisi Kedua, 2018.
- [6] Hussain, S., et al., "Deploying Machine Learning Models: A Comparative Study of Flask, FastAPI, and Streamlit", *International Journal of Advanced Computer Science and Applications*, Vol. 12, No. 5, 2021.
- [7] Zhang, Ying., Xiao, Gang., **Named Entity Recognition Datasets: A Classification Framework**, *International Journal of Computational Intelligence Systems*, Vol. 17, No. 71, Hal. 1-17, Springer, 2024.
- [8] Deußer, Tobias., La., Hillebrand, Lars., Bauckhage, Christian., Sifa, Rafet., **Informed Named Entity Recognition Decoding for Generative Language Models**, *IEEE International Conference on Big Data (BigData)*, Hal. 1-13, 2024.
- [9] Jurafsky, Daniel, Martin, James H., *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Pearson Prentice Hall, Edisi Kedua, 2008.
- [10] Marcondes, Francisco S., Almeida, José João, Novais, Paulo, **Large Language Models: Compilers for the 4th Generation of Programming Languages?**, 12th Symposium on Languages, Applications and Technologies (SLATE 2023), Hal. 10:1-10:8, 2023.
- [11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "**BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, MN: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423
- [12] Mahalakshmi, G., Kumar, A. A., Senthilnayaki, B., Duraimurugan, J., **Job Recommendation System Based On Skill Sets**, *International Journal of Creative Research Thoughts (IJCRT)*, Vol. 10, No. 8, Hal. a770-a785, IJCRT.ORG, 2022.
- [13] Alsaif, Suleiman Ali, Sassi Hidri, Minyar, Eleraky, Hassan Ahmed, Ferjani, Imen, Amami, Rimah, **Learning-Based Matched Representation System for Job Recommendation**, Computers, Vol. 11, No. 161, Hal. 1-18, MDPI, 2022.

- [14] S. A. Alsaif, M. S. Hidri, I. Ferjani, H. A. Eleraky, and A. Hidri, "NLP-Based Bi-Directional Recommendation System: Towards Recommending Jobs to Job Seekers and Resumes to Recruiters," *Big Data and Cognitive Computing*, vol. 6, no. 4, p. 147, Dec. 2022, doi: 10.3390/bdcc6040147.
- [15] Tentua, Meilany Nonsi, Azhari, Azhari, Musdholifah, Aina, **The Multi Agent System for Job Recommendation**, *The 7th International Conference on DV-Xa Method*, Hal. 1-7, 2020.
- [16] Liu, Mengshu, Wang, Jingya, Abdelfatah, Kareem, Korayem, Mohammed, **Tripartite Vector Representations for Better Job Recommendation**, *DI2KG '19*, Anchorage, Alaska, USA, 2019.

LAMPIRAN

Lampiran berisi informasi yang menunjang bahasan dalam Buku Proyek Akhir, seperti data yang cukup banyak, datasheet, pembuktian matematis, dan lain-lain. Lampiran dapat berupa tabel, gambar, worksheet, foto, dan lain-lain. Jika lembaran lampiran lebih besar dari batas ukuran buku proyek akhir, lembar lampiran dapat dilipat sehingga tidak melebihi batas ukuran buku.

Contoh Judul Lampiran