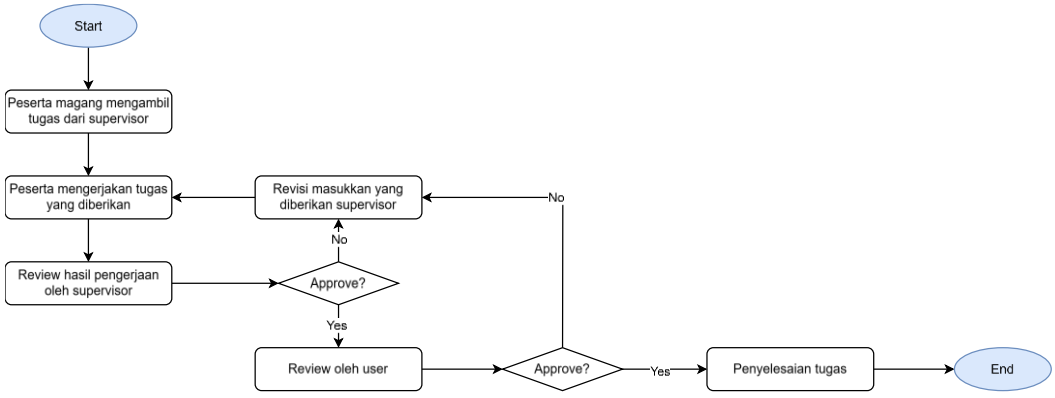
BAB III
PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Posisi magang sebagai *Data Scientist intern* berada pada divisi *Research and Development*. Perusahaan Artha Telekomindo mencari *intern* yang akan diposisikan pada proyek/tim berdasarkan kemampuan yang dimiliki oleh masing-masing intern. Sebagai mahasiswa sistem informasi dengan peminatan *Big Data Analytics* yang befokus pada analisis dan pemrosesan data, posisi sebagai *Data Scientist intern* akan bekerja pada segala tim yang diposisikan untuk penugasan, semisal intern dibutuhkan pada tim *Research and Development* maka intern akan mengerjakan tugasnya hingga selesai, dan dapat bekerja untuk tim lain yang diposisikan selanjutnya. Tugas yang diberikan berkaitan dengan *Artificial Intelligence* perusahaan pada lokasi internal. Detail kedudukan posisi intern pada divisi *Research and Development* dapat dilihat pada gambar 3.1 dibawah.



Gambar 3.1. Alur Kerja Pelaksana Magang

Alur kerja pada gambar 3.1 dimulai dengan peserta magang menerima tugas dari Supervisor, yang tugasnya berasal dari permintaan langsung pengguna *(user)*. Tugas ini biasanya berhubungan dengan riset dan pengembangan sistem baru yang sedang dikembangkan. Supervisor akan memberikan penjelasan mengenai ruang lingkup tugas, tujuan yang ingin dicapai, serta batasan atau kendala yang mungkin dihadapi dalam proses pengerjaan. Selain itu, Supervisor juga akan memberikan referensi atau dokumentasi yang diperlukan untuk membantu peserta magang memahami tugas dengan lebih baik.

Setelah menerima dan memahami arahan yang diberikan, peserta magang mulai mengerjakan tugas sesuai dengan panduan yang telah dijelaskan. Dalam prosesnya, mereka memiliki kesempatan untuk berkoordinasi dan berdiskusi dengan Supervisor apabila menemui kendala teknis atau membutuhkan klarifikasi lebih lanjut mengenai tugas yang sedang dikerjakan. Komunikasi yang baik dalam tahap ini sangat penting untuk memastikan bahwa tugas dikerjakan dengan efisien dan sesuai dengan harapan.

Setelah tugas selesai dikerjakan, hasil pekerjaan akan diperiksa oleh Supervisor melalui tahap *review*. Supervisor akan mengevaluasi apakah hasil yang dihasilkan sudah sesuai dengan standar dan kebutuhan yang telah ditentukan sebelumnya. Evaluasi ini mencakup berbagai aspek, seperti kualitas implementasi, tingkat kesalahan, efektivitas solusi yang diberikan, serta kepatuhan terhadap spesifikasi yang telah disepakati. Jika pekerjaan sudah memenuhi standar yang ditetapkan, maka Supervisor akan menyetujui hasilnya, dan tugas dapat dilanjutkan ke tahap berikutnya. Namun, jika masih terdapat kekurangan atau kesalahan, Supervisor akan memberikan masukan secara mendetail terkait aspek yang perlu diperbaiki. Peserta magang diwajibkan untuk melakukan revisi berdasarkan umpan balik yang diberikan, baik dalam hal perbaikan teknis, penyempurnaan desain, atau optimalisasi kinerja sistem. Revisi ini dilakukan hingga Supervisor memberikan persetujuan penuh terhadap hasil kerja peserta magang.

Apabila Supervisor sudah menyetujui hasil revisi, tugas akan diteruskan ke tahap berikutnya, yaitu proses *review* oleh pengguna (*user*). Pengguna akan menilai apakah hasil pekerjaan sudah sesuai dengan kebutuhan bisnis dan ekspektasi mereka. Proses ini biasanya melibatkan pengujian langsung terhadap sistem atau fitur yang telah dikembangkan. Jika pengguna menemukan aspek yang masih perlu diperbaiki atau disesuaikan, mereka akan memberikan umpan balik yang harus segera ditindaklanjuti oleh peserta magang dalam bentuk revisi tambahan. Setelah mendapatkan persetujuan dari pengguna, tugas akan memasuki tahap penyelesaian akhir. Pada tahap ini, peserta magang akan melakukan perbaikan lebih lanjut untuk meningkatkan kualitas hasil pekerjaan sebelum sistem atau fitur tersebut diterapkan

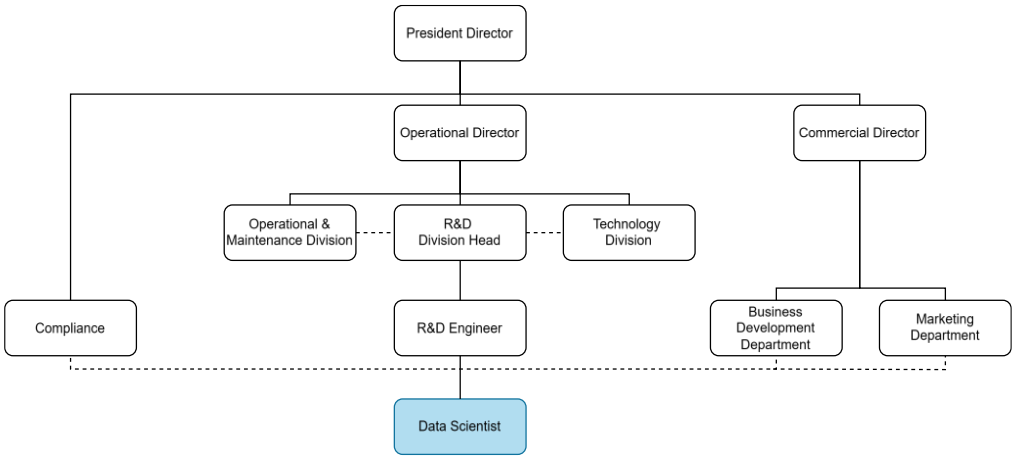
sepenuhnya. Perbaikan ini dapat mencakup penyempurnaan tampilan agar lebih *user-friendly,* peningkatan akurasi sistem untuk meningkatkan performa, serta dokumentasi tambahan agar sistem dapat lebih mudah dipahami dan digunakan oleh pengguna di masa mendatang. Selain itu, peserta magang juga dapat melakukan pengujian tambahan untuk memastikan bahwa sistem berfungsi dengan optimal sebelum diserahkan sepenuhnya kepada pengguna.

Sehabis semua tahap penyempurnaan selesai dan tugas telah memenuhi standar yang diharapkan oleh Supervisor serta pengguna, maka proses pengerjaan tugas dapat dinyatakan selesai. Dokumentasi terkait hasil pekerjaan akan disusun sebagai laporan akhir untuk mendukung keberlanjutan proyek atau sebagai referensi bagi tim yang akan mengembangkan sistem lebih lanjut di masa depan. Dengan demikian, proses kerja yang telah dilakukan oleh peserta magang dapat memberikan kontribusi nyata dalam pengembangan sistem yang lebih baik dan sesuai dengan kebutuhan organisasi.

3.1.1 Kedudukan

Divisi Research and Development (R&D) di PT Artha Telekomindo berada di bawah kepemimpinan Bapak Antonius Wisanggeni yang menjabat sebagai Research and Development Division Head. Dalam struktur organisasi divisi ini, Bapak Taufiq Ahmadi memiliki tanggung jawab sebagai mentor dan pembimbing langsung bagi para peserta magang yang ditempatkan pada posisi Data Scientist Intern. Selain mendapatkan arahan langsung dari kepala divisi, setiap peserta magang juga memperoleh supervisi khusus dari staf profesional yang sesuai dengan posisi dan proyek yang ditugaskan.

Dalam konteks pelaksanaan magang di divisi Research and Development, peserta magang menempati posisi sebagai Data Scientist Intern dan bekerja di bawah bimbingan langsung Bapak Taufiq Ahmadi, yang juga menjabat sebagai Data Scientist di perusahaan. Selama menjalankan kegiatan magang, peserta diberikan serangkaian tugas dan tanggung jawab yang dirancang untuk meningkatkan pemahaman serta keterampilan teknis di bidang Data Science, sekaligus memberikan kontribusi nyata terhadap kebutuhan dan pengembangan sistem internal Perusahaan.



Gambar 3.2. Kedudukan Peserta Magang

Tanggung jawab utama yang diemban oleh peserta magang selama masa praktik kerja antara lain:

1. Merancang dan mengembangkan sistem berbasis *Artificial Intelligence* (AI) untuk mendukung proses bisnis perusahaan.

2. Melakukan proses *data cleaning* atau pembersihan terhadap data internal perusahaan guna memastikan akurasi dan konsistensi data yang akan digunakan dalam analisis dan pemodelan.

3. Membangun model-model *AI* dan *Machine Learning* yang dapat diintegrasikan ke dalam sistem perusahaan untuk meningkatkan efisiensi dan efektivitas kerja.

4. Meningkatkan kualitas dan efektivitas pengelolaan data perusahaan melalui penerapan metode analisis data yang sesuai dan berbasis teknologi.

5. Memelihara dan memastikan ketersediaan layanan *Application Programming Interface* (API) yang digunakan oleh sistem internal maupun eksternal perusahaan.

6. Menyusun dokumentasi teknis yang berkaitan dengan proses pengembangan model *AI*, termasuk dokumentasi alur kerja, penggunaan data, metode pelatihan model, serta hasil evaluasi performa sistem.

Seluruh tugas dan proyek ini diberikan secara bertahap oleh supervisor magang yang telah ditunjuk oleh pihak perusahaan, dan dikerjakan berdasarkan prioritas kebutuhan yang ada di divisi *Research and Development*. Dengan menjalankan tanggung jawab tersebut, peserta magang diharapkan dapat memperoleh pengalaman praktis yang bernilai sekaligus memberikan kontribusi positif bagi perusahaan.

3.2 Tugas dan Uraian Kerja Magang

Dalam pelaksanaan magang di PT. Artha Telekomindo, divisi *Research and Development* sebagai *Data Scientist (Intern)* menjalankan berbagai tugas yang berkaitan dengan bidang big data, khususnya pada pengembangan efektivitas perusahaan dalam mencari pekerja baru, sesuai dengan peminatan peserta magang yaitu *Artificial Intelligence*. Tugas-tugas ini bertujuan untuk memberikan pengetahuan dan pengalaman lebih dalam kepada peserta magang. Selain itu, *Data Scientist (intern)* juga terlibat dalam proyek-proyek yang diberikan oleh supervisor dan tim di divisi *Research and Development*. Uraian kerja ini mencakup tanggung jawab peserta magang selama periode magang di PT. Artha Telekomindo.

3.2.1 Tugas Kerja Magang

Selama menjalani praktik kerja magang di PT. Artha Telekomindo sebagai *Data Scientist (Intern)*, peserta magang telah menyelesaikan beragam tugas. Tugas-tugas ini mencakup pengembangan efektivitas perusahaan

dengan bantuan AI. Tabel 3.1. menampilkan daftar tugas-tugas yang dilakukan oleh peserta magang selama berada di perusahaan.

Tabel 3.1. Tabel Tugas Kerja Magang

No.	Kegiatan	Kordinasi	Pelaksanaan
1.	Webscraping X comments on AI		
1.1	*Analyzing web scraping data*	*Research and Development Manager*	4 Februari 2025 – 5 Februari 2025
1.2	*Prepare the web scraping data*		4 Februari 2025 – 5 Februari 2025
1.3	*Implement visual based on what is needed*		4 Februari 2025 – 5 Februari 2025
2.	*CV Reviewer Bot*		
2.1	*Understanding what user needs*	*Human Resource Development Team*	6 Februari 2025 – 9 Februari 2025
2.2	*Brainstorming ideas with Supervisor*		6 Februari 2025 – 9 Februari 2025
2.3	*Implementing ideas to code using deepseek*		6 Februari 2025 – 9 Februari 2025
2.4	*Enhancing the process*		6 Februari 2025 – 9 Februari 2025
2.5	*Trying other model such as Sahabat and llama*		6 Februari 2025 – 9 Februari 2025
2.6	*Intergrate MongoDB for easy access*		6 Februari 2025 – 9 Februari 2025
3.	*Document Reader and analyser*		
3.1	*Understanding how LLm works*	*Research and Development Manager*	10 Februari 2025 – 18 Februari 2025
3.2	*Create a knowledge from a document*		19 Februari 2025 – 28 Februari 2025
3.3	*Enhancing the needed feature*		3 Maret 2025 – 13 Maret 2025
3.4	*Impelement knowledge from database*		14 Maret 2025 – 21 Maret 2025
3.5	*Integrate chromadb to make the analyzer better.*		24 Maret 2025 – 31 Maret 2025
4.	*Chatbot with Knowledge*		
4.1	*Understanding the problem*	*Research and Development Manager*	1 April 2025 – 4 April 2025
4.2	*Making the knowledge using Document to text*		7 April 2025 – 9 April 2025
4.3	*Integrating the Bot*		10 April 2025 – 16 April 2025
4.4	*Making the bot with the knowledge*		17 April 2025 – 24 April 2025
4.5	*Enhance analytics result*		25 April 2025 – 30 April 2025

No.	Kegiatan	Kordinasi	Pelaksanaan
4.	*Chatbot with Knowledge*		
4.1	*Understanding the problem*	*Research and Development Manager*	1 April 2025 – 4 April 2025
4.2	*Making the knowledge using Document to text*		7 April 2025 – 9 April 2025
4.3	*Integrating the Bot*		10 April 2025 – 16 April 2025
4.4	*Making the bot with the knowledge*		17 April 2025 – 24 April 2025
4.5	*Enhance analytics result*		25 April 2025 – 30 April 2025
5.	*Model creation (Speech to text)*		
5.1	*Understanding the problem*	*Research and Development Manager*	1 Mei 2025 – 2 Mei 2025
5.2	*Learning to make model using h5*		2 Mei 2025 – 7 Mei 2025
5.3	*Model training*		8 Mei 2025 – 14 Mei 2025
5.4	*Optimizing the model*		15 Mei 2025 – 20 Mei 2025
5.5	*Implement model*		21 Mei 2025 – 26 Mei 2025
5.6	*Enhance model result*		27 Mei 2025 – 30 Mei 2025

Tabel 3.1 berisi daftar tugas yang dikerjakan selama magang di PT Artha Telekomindo dari Februari hingga Mei 2025. Tugas-tugas tersebut meliputi *web scraping* data AI, pembuatan sistem *CV reviewer*, pengembangan pembaca dokumen berbasis LLM, pembuatan chatbot internal, dan model klasifikasi suara untuk membedakan suara HR dan kandidat. Seluruh kegiatan dilakukan secara bertahap dan dikoordinasikan dengan tim *Research and Development* serta *Human Resource.*

3.2.2 Uraian Kerja Magang

Peserta magang melaksanakan praktik kerja magang selama total 640 jam di PT Artha Telekomindo, dengan kegiatan yang diarahkan dan dibimbing langsung oleh *Research and Development Manager* serta *Human Resource Development Manager*. Selama periode magang, peserta mendapatkan beberapa tugas utama yang berkaitan langsung dengan pengembangan sistem berbasis *Artificial Intelligence (AI)* dan pengolahan data. Salah satu tugas yang menjadi bagian penting dalam proses pembelajaran dan kontribusi peserta magang adalah proyek *web scraping*, yang dijelaskan sebagai berikut:

3.2.2.1 Web Scraping

Praktik kerja magang dilaksanakan selama total 640 jam di PT Artha Telekomindo. Selama periode tersebut, kegiatan diarahkan dan dibimbing langsung oleh Research and Development Manager serta Human Resource Development Manager. Beberapa tugas utama yang diberikan berkaitan langsung dengan pengembangan sistem berbasis Artificial Intelligence (AI) dan pengolahan data. Salah satu tugas penting dalam kegiatan tersebut adalah proyek *web scraping*, yang dijelaskan sebagai berikut:

Tugas web scraping diberikan untuk mengumpulkan data secara otomatis dari internet, dengan fokus utama pada platform X (sebelumnya dikenal sebagai Twitter). Tujuan dari kegiatan ini adalah mengidentifikasi tren terkini dalam bidang Artificial Intelligence berdasarkan opini publik yang tersebar di media sosial.

Secara umum, web scraping merupakan teknik otomatisasi untuk menyalin dan mengumpulkan informasi dari halaman situs web. Informasi yang diperoleh digunakan sebagai dasar analisis sentimen, pemetaan tren, serta visualisasi data yang dapat mendukung pengambilan keputusan strategis dalam pengembangan teknologi AI di perusahaan.

Tahap awal dari proyek ini adalah analisis terhadap data mentah hasil scraping. Data diambil dari platform terbuka yang menyediakan informasi aktual terkait diskusi publik mengenai topik *Artificial Intelligence*.

Dengan bimbingan dari *supervisor*, dilakukan eksplorasi awal terhadap konten teks yang dikumpulkan, dengan fokus pada identifikasi topik, kata kunci, atau teknologi yang sedang menjadi perhatian publik. Proses ini bertujuan memberikan gambaran awal mengenai persepsi dan minat masyarakat terhadap perkembangan AI.

Cuplikan data mentah hasil scraping ditampilkan pada Gambar 3.3, yang mencakup kolom seperti full_text, created_at, user_location, dan hashtags.

conversation_id_str	created_at	favorite_count	full_text	id_str	image_url	in_reply_to_screen	lang	location	quote_count	reply_count	retweet_count	tweet_url	user_id_str	username
1.88E+18	Mon Feb 03 02:01:22 +0000 2025	0	@Telnium1 the deep term has been	1.88E+18		Telnium1	en	married	0	0	0	https://x.com/ilwra	1602580286	liverandheart
1.88E+18	Mon Feb 03 02:01:07 +0000 2025	0	South Korea s privacy watchdog pl	1.88E+18			en	Hong Kong and Ta	0	0	0	https://x.com/dnyst	454602286	dnystedt
1.88E+18	Mon Feb 03 02:01:06 +0000 2025	1	Dall e Mini is open source. Like Lim	1.88E+18			en	The Internet	0	0	0	https://x.com/dabad	1.32224E+18	abaddonbrain
1.88E+18	Mon Feb 03 02:01:05 +0000 2025	0	Exploring the Benefits of a #Custom	1.88E+18	https://pbs.twimg.com/media/GI0_S5A	en	Prague, Czech Rep	0	0	0	https://x.com/RiteK	235439535	RiteKitAPI
1.88E+18	Mon Feb 03 02:01:03 +0000 2025	0	o3 and DeepSeek-R1 are changing	1.88E+18		QianAITech	en	Join 259 Readers E	0	0	0	https://x.com/Qian	1.72169E+18	QianAITech
1.88E+18	Mon Feb 03 02:01:00 +0000 2025	0	@kevinzou Theyre willing to open s	1.88E+18		JChengWSJ	en	Beijing	0	0	0	https://x.com/JChe	382789364	JChengWSJ
1.88E+18	Mon Feb 03 02:00:55 +0000 2025	0	DeepSeek China OpenAI AI R1 AI vi	1.88E+18			en		0	0	0	https://x.com/randi	1.32639E+18	random_pure
1.88E+18	Mon Feb 03 02:00:43 +0000 2025	0	@iAnonPatriot DeepSeek is eating	1.88E+18		iAnonPatriot	en	Colorado, USA	0	0	0	https://x.com/wave	60591188	wave_cometh
1.88E+18	Mon Feb 03 02:00:42 +0000 2025	0	Damn DeepSeek R1 needs to chill f	1.88E+18	https://pbs.twimg.com/media/GI0-Tes	en	Toronto, Canada	0	0	0	https://x.com/Sully	1.55639E+18	SullyBillions
1.88E+18	Mon Feb 03 02:00:38 +0000 2025	0	Business News for February 03 202	1.88E+18			en	undisclosed locatic	0	0	0	https://x.com/news	1.67819E+18	news2_website
1.88E+18	Mon Feb 03 02:00:34 +0000 2025	0	@CryptoWendyO Xi Declined meeti	1.88E+18		CryptoWendyO	en	Texas, USA	0	0	0	https://x.com/CPov	1.1654E+18	CPomp75664
1.88E+18	Mon Feb 03 02:00:23 +0000 2025	0	SANS ISC Stormcast Feb 3rd 2025	1.88E+18	https://pbs.twimg.com/media/GI0_Ipub	en	Jacksonville, FL, U	0	0	0	https://x.com/sans_	6821102	sans_isc
1.88E+18	Mon Feb 03 02:00:23 +0000 2025	0	@OpenAI Copy paste of DeepSeek	1.88E+18		OpenAI	en		0	0	0	https://x.com/chick	1.27895E+18	chickVanessa878
1.88E+18	Mon Feb 03 02:00:18 +0000 2025	0	OpenAI debuts Operator an AI ager	1.88E+18			en		0	0	0	https://x.com/AIAg	1.652E+18	AIAgentOne
1.88E+18	Mon Feb 03 02:00:10 +0000 2025	0	@scamai thanks to deepseek indeec	1.88E+18		sama	en	63.403367, -15.047	0	0	0	https://x.com/JeffHi	2583163524	JeffreyH630
1.88E+18	Mon Feb 03 02:00:09 +0000 2025	1	He s not been seen on a yacht with	1.88E+18			en	London	0	0	0	https://x.com/FTIst	16949452	FT
1.88E+18	Mon Feb 03 02:00:05 +0000 2025	0	Sam Altman says OpenAI is on the	1.88E+18			en	Houston, TX	0	0	0	https://x.com/STEV	82769852	STEVEPMP
1.88E+18	Mon Feb 03 02:00:03 +0000 2025	0	#DeepSeek DeepThink-R1 The dar	1.88E+18	https://pbs.twimg.com/media/GI0-vaS	en		0	0	0	https://x.com/name	1.87E+18	name535761952351
1.88E+18	Mon Feb 03 02:00:01 +0000 2025	0	According to Pandora Tech News h	1.88E+18	https://pbs.twimg.com/media/GIVvEVI	en		0	0	0	https://x.com/impai	1.8784E+18	impandoratech

Gambar 3.3. Data *Web Scraping*

Setelah data dikumpulkan, proses selanjutnya adalah data *preparation*, yakni tahap persiapan data agar layak untuk dianalisis lebih lanjut. Tahapan ini meliputi:

1. Pembersihan *data* (*data cleaning*): menghapus data duplikat, memperbaiki kesalahan ejaan, menghilangkan simbol yang tidak relevan, dan menyamakan format teks.
2. Pelabelan sentimen (*sentiment labeling*): mengklasifikasikan setiap entri teks ke dalam kategori sentimen positif, negatif, atau netral.
3. Perhitungan *hashtag*: menghitung frekuensi kemunculan setiap *hashtag* yang terdapat dalam teks.

Tujuan utama dari tahap ini adalah memastikan bahwa data yang akan dianalisis bersih, terstruktur, dan representatif terhadap kondisi sebenarnya. Data yang tidak melalui proses persiapan dengan baik berpotensi menghasilkan analisis yang tidak akurat dan menyesatkan.

```
dt2 = ['created_at', 'full_text', 'location']
data = data[dt2]

data.head(5)
```

Gambar 3.4. *Data Preparations*

Visualisasi alur *data preparation* ditampilkan pada Gambar 3.4, sementara hasil pelabelan sentimen terhadap kolom *full_text* ditampilkan pada Gambar 3.5. Dari hasil pelabelan ini, dapat diketahui proporsi opini masyarakat terhadap isu-isu yang berkaitan dengan AI.

```
from textblob import TextBlob

def get_sentiment(text):
    blob = TextBlob(text)
    polarity = blob.sentiment.polarity

    if polarity > 0:
        sentiment_category = "positive"
    elif polarity < 0:
        sentiment_category = "negative"
    else:
        sentiment_category = "neutral"

    return polarity, sentiment_category

data[['sentiment', 'sentiment_category']] = data['full_text'].apply(lambda x: pd.Series(get_sentiment(x)))

data[['created_at', 'full_text', 'location', 'sentiment', 'sentiment_category']].head(10)
```

Gambar 3.5. Sentimen Data

Selain itu, dilakukan pula penghitungan terhadap jumlah kemunculan setiap *hashtag* dalam *dataset*. Hasil ini memberikan gambaran tentang topik-topik yang sedang populer. Semakin sering sebuah *hashtag* muncul, semakin besar kemungkinan bahwa topik tersebut sedang ramai diperbincangkan. Hasil penghitungan ini divisualisasikan pada Gambar 3.6.

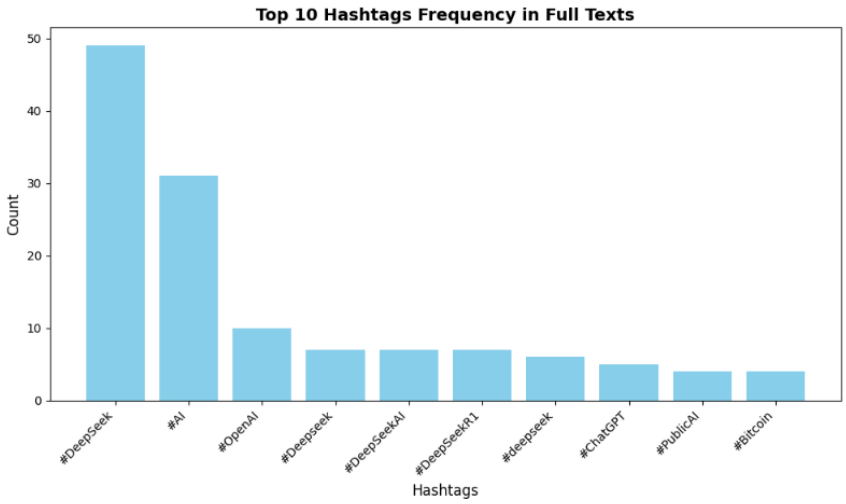
```
hashtags, counts = zip(*hashtag_counts.items())

plt.figure(figsize=(10, 6))
plt.bar(hashtags, counts, color='skyblue')

plt.xlabel('Hashtags', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.title('Hashtag Frequency in Full Texts', fontsize=14, fontweight='bold')

plt.xticks(rotation=45, ha='right')

plt.tight_layout()
plt.show()
```

Gambar 3.6. *Hashtags Count*

Tahapan akhir dari proyek *web scraping* adalah visualisasi data yang telah melalui proses pembersihan dan persiapan sebelumnya. Visualisasi ini memiliki peran penting dalam menyampaikan informasi

secara ringkas, mudah dipahami, dan mendukung pengambilan keputusan berbasis data.

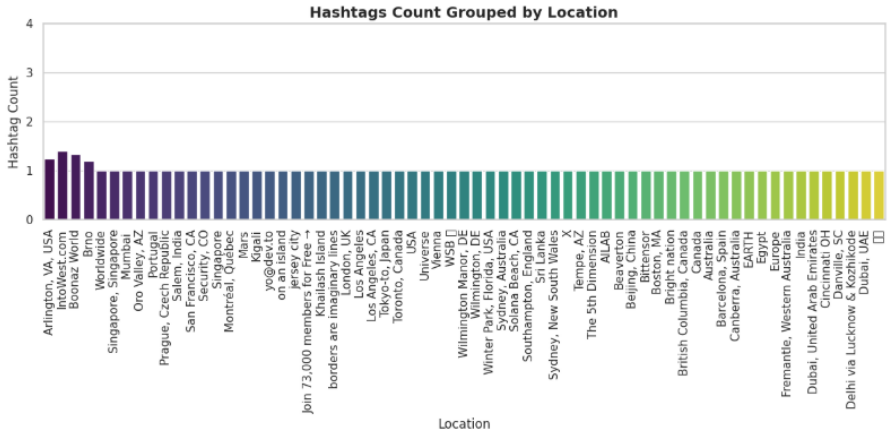
Visualisasi pertama dibuat dalam bentuk diagram batang (*bar chart*) yang ditampilkan pada Gambar 3.7, menggambarkan sepuluh *hashtag* terpopuler yang berkaitan dengan AI berdasarkan frekuensi kemunculannya. Dari grafik tersebut, terlihat bahwa *hashtag* seperti #deepseek, #AI, dan #OpenAI merupakan yang paling sering dibahas. Informasi ini penting untuk memahami minat masyarakat terhadap teknologi tertentu dan menjadi masukan strategis bagi perusahaan dalam menentukan fokus pengembangan produk atau layanan berbasis AI.



Gambar 3.7. *Barchart Hashtags Count*

Selanjutnya, Gambar 3.8 menyajikan visualisasi tambahan berupa diagram batang yang menghubungkan frekuensi hashtag dengan lokasi geografis pengguna. Tujuan dari visualisasi ini adalah untuk memahami bagaimana persebaran diskusi mengenai AI tersebar di berbagai wilayah. Informasi ini dapat dimanfaatkan oleh perusahaan untuk menentukan target pasar, merancang kampanye edukasi berbasis

wilayah, serta mengidentifikasi peluang kolaborasi di daerah dengan tingkat diskusi AI yang tinggi.



Gambar 3.8. ***Barchart Hashtags/Locations***

3.2.2.2 CV Reviewer Bot

Tugas ini merupakan salah satu tanggung jawab utama dalam program magang, yaitu merancang dan mengembangkan sistem berbasis *Artificial Intelligence* (AI) yang bertujuan untuk membantu tim Human Resource Development (HRD) dalam mempercepat sekaligus meningkatkan akurasi proses rekrutmen karyawan baru.

Sistem yang dikembangkan dirancang untuk menganalisis *Curriculum Vitae* (CV) para pelamar dan menilai tingkat kecocokan kandidat terhadap kriteria yang telah ditentukan oleh perusahaan. Dengan keberadaan sistem ini, proses penyaringan awal dapat dilakukan secara otomatis, sehingga menghemat waktu dan tenaga tim HRD serta mengurangi risiko subjektivitas dalam proses penilaian awal.

Tahapan awal dimulai dengan proses diskusi intensif bersama supervisor dan tim HRD guna memperoleh pemahaman menyeluruh terhadap kebutuhan pengguna. Dalam tahap ini, dilakukan identifikasi

terhadap komponen-komponen utama yang perlu dianalisis dalam sebuah CV, seperti:

1. *Hard skills* (keterampilan teknis)
2. Pengalaman kerja yang relevan
3. Latar belakang pendidikan
4. Sertifikasi atau penghargaan
5. Bahasa yang digunakan

Tujuan dari proses ini adalah memastikan bahwa sistem yang dikembangkan mampu menyaring informasi yang benar-benar relevan dengan standar dan kebutuhan perekrutan perusahaan.

Setelah pemetaan kebutuhan pengguna selesai dilakukan, dilakukan sesi *brainstorming* bersama supervisor untuk menentukan pendekatan teknis yang paling sesuai. Berdasarkan hasil analisis tren dari proyek *web scraping* sebelumnya, ditemukan bahwa model bahasa besar DeepSeek memiliki performa tinggi dalam bidang *Natural Language Processing* (NLP). Oleh karena itu, disepakati untuk menggunakan model DeepSeek R1 Distilled 7B sebagai inti sistem analisis CV.

Pada tahap ini, dilakukan implementasi sistem menggunakan model DeepSeek R1 Distilled 7B yang tersedia melalui platform Hugging Face. Model ini digunakan untuk membaca, memahami, dan

menganalisis isi CV secara otomatis berdasarkan konteks dan isi teks. Alur kerja implementasi ditampilkan pada Gambar 3.9.

```
model_name = "deepseek-ai/DeepSeek-R1-Distill-Qwen-7B"
tokenizer = AutoTokenizer.from_pretrained(model_name)

model = AutoModelForCausalLM.from_pretrained(
    model_name,
    torch_dtype=torch.bfloat16,
    device_map="auto",
    offload_folder="offload"
)
```

Gambar 3.9. ***Deepseek Implementation***

Untuk menunjang proses implementasi sistem, digunakan beberapa pustaka (*libraries*) penting yang berperan dalam berbagai tahapan pengembangan, antara lain: pdfplumber yang digunakan untuk mengekstrak teks dari dokumen PDF; pymongo yang berfungsi menghubungkan sistem dengan basis data MongoDB; transformers, khususnya modul AutoTokenizer dan Torch, yang digunakan untuk menjalankan model DeepSeek; serta tkinter, yang dimanfaatkan untuk membangun antarmuka pengguna (*user interface*) yang sederhana dan mudah digunakan. Seluruh pustaka yang digunakan dalam proyek ini ditampilkan pada Gambar 3.10.

```
import torch
import os
import pdfplumber
from tkinter import Tk, filedialog
from transformers import AutoTokenizer, AutoModelForCausalLM, GenerationConfig
from pymongo import MongoClient
```

Gambar 3.10. ***Import Libraries***

Selama proses pengembangan, dilakukan berbagai penyempurnaan teknis untuk meningkatkan kualitas output sistem. Beberapa bentuk optimasi yang dilakukan meliputi:

1. Penyesuaian jumlah maksimum token yang dapat diproses oleh model.
2. Perbaikan struktur prompt agar lebih terarah, spesifik, dan sesuai dengan konteks rekrutmen.

Dengan melakukan penyempurnaan prompt engineering, sistem menjadi lebih mampu memahami konteks dan menghasilkan analisis yang lebih relevan. Contoh format prompt yang digunakan ditampilkan pada Gambar 3.11.

```python
# Match qualifications using LLM prompt
def match_qualifications(cv_text, job_qualifications):
    if not job_qualifications:
        return 0

    messages = [
        {"role": "system", "content": f"""
        You are an AI assistant specialized in evaluating job qualifications.
        The following job qualifications are required for the position:

        {job_qualifications}

        Analyze the provided CV and determine if the candidate meets these qualifications.
        Respond with a number score (0-10) indicating the match level, where:
        - 0 means no match at all.
        - 10 means a perfect match.
        """},
        {"role": "user", "content": f"CV content:\n{cv_text}"}
    ]
```

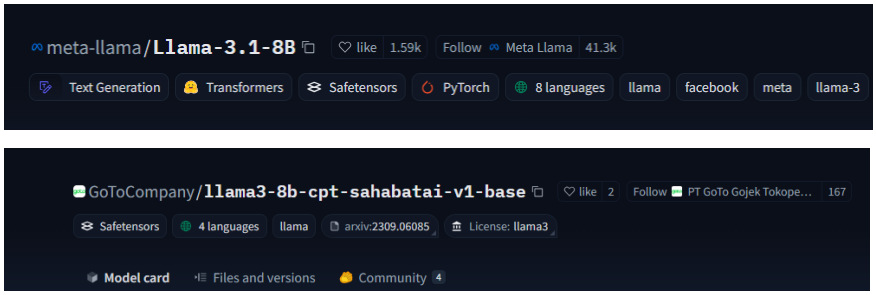
Gambar 3.11. *Prompting*

Untuk mengevaluasi efektivitas sistem dilakukan beberapa eksperimen menggunakan beberapa model alternatif seperti:

1. Sahabat AI (oleh GoTo)
2. LLaMA (oleh Meta)

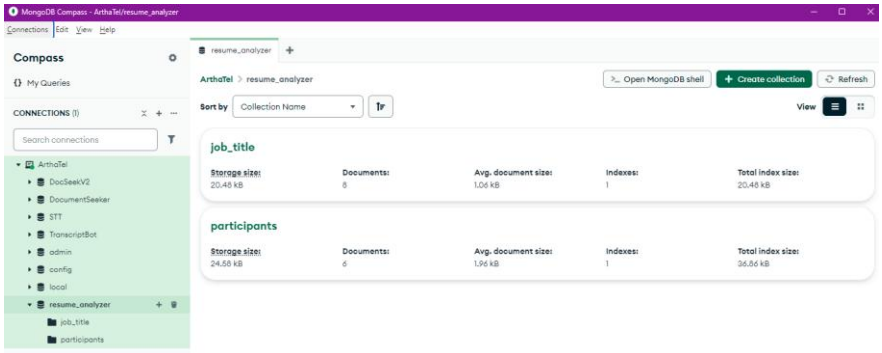
Tujuan dari eksperimen ini adalah untuk membandingkan performa antar model, khususnya dalam aspek keakuratan, keterbacaan, dan relevansi hasil analisis. Berdasarkan pengujian, diketahui bahwa model DeepSeek masih memberikan hasil yang paling terstruktur dan

relevan, sedangkan Sahabat AI dan LLaMA menghasilkan keluaran yang kurang terarah dan kurang sesuai dengan kebutuhan proses rekrutmen. Hasil perbandingan ditampilkan pada Gambar 3.12



Gambar 3.12. *Model* Lainnya

Tahap akhir dari pengembangan sistem adalah integrasi dengan basis data MongoDB. Tujuan dari integrasi ini adalah agar seluruh hasil analisis dapat tersimpan secara sistematis dan dapat diakses kembali oleh tim HRD dengan mudah. Penyimpanan data ke dalam MongoDB memungkinkan tim untuk melacak riwayat evaluasi kandidat, melakukan pencarian cepat, serta mengarsipkan hasil penilaian untuk keperluan audit atau analisis lanjutan. Alur integrasi sistem dengan MongoDB ditampilkan pada Gambar 3.13.



Gambar 3.13. MongoDB

3.2.2.3 Document Reader and Analyzer

Tugas ini merupakan salah satu proyek yang dipercayakan untuk mendukung efisiensi kerja tim IT Security dalam menganalisis dokumen pelanggan. Berdasarkan arahan dari supervisor tugas ini dirancang untuk membangun sebuah sistem berbasis *Artificial Intelligence (AI)* yang memiliki kemampuan untuk membaca, memahami, dan mengekstrak informasi penting dari dokumen internal perusahaan secara otomatis.

Sistem ini bertujuan untuk membantu tim dalam mengidentifikasi layanan apa saja yang digunakan oleh pelanggan, serta mendukung proses pengambilan keputusan yang cepat, tepat, dan berbasis data.

Langkah pertama yang dilakukan dalam proses pengembangan sistem ini adalah memahami cara kerja *Large Language Model (LLM)*, karena data mentah yang diberikan oleh supervisor tersedia dalam format Excel yang bersifat tabular.

Sebagaimana diketahui, LLM lebih optimal dalam menganalisis data berbasis teks tak terstruktur (*unstructured text*) dibandingkan dengan data *tabular*. Oleh karena itu, perlu dilakukan transformasi terhadap data tersebut agar dapat dipahami dan diproses secara efektif oleh model.

Gambar 3.14 memperlihatkan format awal data pelanggan yang masih berbentuk tabel Excel. Transformasi data ini menjadi landasan penting bagi keseluruhan sistem, karena menentukan bagaimana model dapat memahami konteks dan hubungan antar informasi dalam dokumen

Username	Domain	Hosting_Location_Domain	Hosting_Location_IP	Services
zhape928	zhape928.co.id	emerald.zhape928.co.id	424.83.75.59	Webhosting, Email
yqifq412	yqifq412.sch.id	sapphire.yqifq412.sch.id	216.30.71.82	Webhosting, Email
dzlyr334	dzlyr334.com	emerald.dzlyr334.com	924.58.16.89	Webhosting, Email
nmhgq598	nmhgq598.sch.id	diamond.nmhgq598.sch.id	176.75.14.13	Webhosting, Email
uvltz576	uvltz576.com	aquamarine.uvltz576.com	492.98.57.89	Webhosting, Email
alqgy309	alqgy309.co.id	opal.alqgy309.co.id	801.54.99.38	Webhosting, Email
corvy591	corvy591.com	topaz.corvy591.com	542.23.25.54	Webhosting, Email
pvwrm444	pvwrm444.sch.id	opal.pvwrm444.sch.id	507.64.77.79	Webhosting, Email
omrdg632	omrdg632.sch.id	diamond.omrdg632.sch.id	830.77.66.61	Webhosting, Email

Gambar 3.14. Data Tabel

Meskipun format Excel cukup terstruktur, model LLM open-source yang digunakan tidak mendukung proses *fine-tuning* secara langsung. Oleh karena itu, data perlu dikonversi menjadi bentuk teks agar bisa dianalisis lebih baik oleh LLM. Proses konversi ini dilakukan dengan mengubah baris data menjadi kalimat informatif yang memuat informasi penting dari setiap entitas pelanggan.

```
text_knowledge = []
for _, row in df.iterrows():
    knowledge_entry = (
        f"User {row['username']} has a domain named '{row['domain']}', "
        f"which is hosted at the location '{row['hosting_location_domain']}' with an IP address of {row['hosting_location_ip']}. "
        f"The services they are utilizing include: {row['services']}. "
    )
    text_knowledge.append({"knowledge": knowledge_entry})

return text_knowledge
```

Gambar 3.15. *Document text-based*

Sebagai contoh, data yang semula berupa kolom “domain”, “IP address”, dan “layanan” diubah menjadi kalimat seperti:"*zhape928 has domain named zhape928.co.id, which is hosted at the location emerald.zhape928.co.id with an IP address of 428.83.75.59. The services they are utilizing include: Webhosting, Email.*"

Transformasi ini ditunjukkan pada Gambar 3.15. Proses ini penting agar model AI dapat memahami relasi antar informasi dalam dokumen dan memberikan respons yang lebih akurat ketika digunakan dalam sistem tanya jawab *(question-answering).*

Selama proses pengembangan, dilakukan penambahan fitur untuk meningkatkan relevansi jawaban dari model AI. Salah satu fitur yang

dikembangkan adalah pengaturan *temperature* dan *prompt engineering*, yang bertujuan untuk mengontrol tingkat variasi dan kreativitas dalam respons yang dihasilkan oleh model.

```python
def generate_thoughtful_response(query, knowledge_base):

    prompt = f"User query: {query}\n\nHere is the entire knowledge base that may help in answering the question:\n"
    prompt += "\n".join(knowledge_base)
    prompt += "\n\nPlease generate a thoughtful response based on the provided information."


    try:
        generator = pipeline("text-generation", model=model, tokenizer=tokenizer)

        generated_response = generator(
            prompt,
            max_new_tokens=150,
            truncation=True,
            return_full_text=False
        )

        return generated_response[0]['generated_text'].strip()

    except Exception as e:
        print(f"Error generating response: {e}")
        return "Sorry, I couldn't generate a response right now."
```

Gambar 3.16. *Prompting*

Penyesuaian ini dilakukan agar model dapat memberikan jawaban yang sesuai dengan konteks dokumen dan kebutuhan tim IT Security. Contoh pengaturan prompt ditampilkan pada Gambar 3.16, yang memperlihatkan bagaimana perintah disusun agar hasil keluaran lebih terarah dan bermakna.

Agar informasi yang telah diekstraksi dari dokumen dapat digunakan secara berkelanjutan, maka seluruh hasil analisis disimpan dalam basis data MongoDB. Informasi yang disimpan meliputi struktur organisasi, daftar layanan, alamat IP, hingga peraturan internal perusahaan. Penyimpanan ini bertujuan agar sistem dapat digunakan sebagai *knowledge base* yang dapat diakses kembali secara otomatis oleh sistem analisis atau *chatbot*.

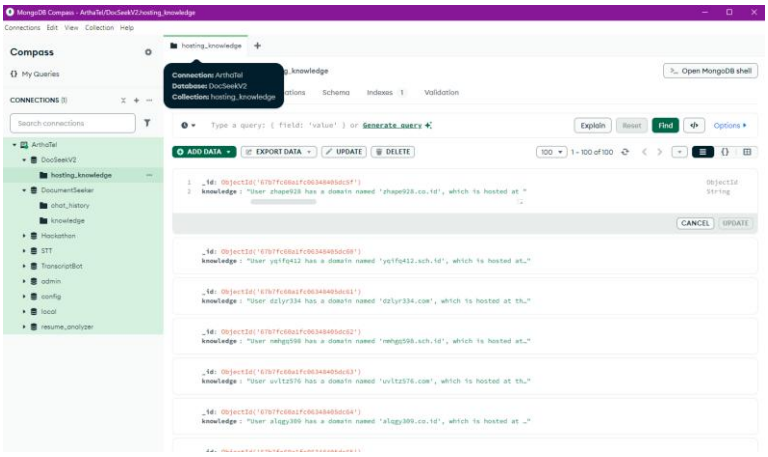
Gambar 3.17 menunjukkan alur penyimpanan pengetahuan dari dokumen ke dalam sistem basis data.

```python
def load_knowledge_base():
    db = connect_mongo()
    knowledge_base = []
    if db is not None:  # Ensure db is not None before proceeding
        try:
            collection = db["hosting_knowledge"]
            knowledge_docs = collection.find()
            for doc in knowledge_docs:
                knowledge_base.append(doc['knowledge'])
        except Exception as e:
            print(f"Error loading knowledge base: {e}")
    return knowledge_base
```

Gambar 3.17. ***Knowledge Gathering***

Untuk meningkatkan performa sistem dalam pencarian informasi, MongoDB diintegrasikan dengan teknik *vector embedding*. Pendekatan ini memungkinkan setiap informasi disimpan dalam bentuk vektor numerik, sehingga pencarian dapat dilakukan berdasarkan kemiripan konteks (bukan sekadar kata kunci). Hal ini mempercepat dan mempermudah sistem dalam menemukan jawaban yang relevan atas pertanyaan pengguna.

Gambar 3.18 menampilkan arsitektur integrasi antara MongoDB dan sistem analisis berbasis LLM, di mana proses *vectorization* dilakukan secara otomatis pada saat data dimasukkan ke dalam sistem.



Gambar 3.18. MongoDB ***Integration***

3.2.2.4 Chatbot With Knowledge

Tugas berikutnya adalah mengembangkan sebuah *chatbot agent* berbasis *Artificial Intelligence (AI)* yang mampu menjawab pertanyaan pengguna berdasarkan pengetahuan yang diperoleh dari dokumen internal perusahaan. Pengetahuan tersebut diambil dari *buku pedoman perusahaan* yang memuat informasi penting seperti struktur organisasi, peraturan internal, alur kerja, hingga prosedur operasional yang berlaku.

Tujuan utama dari pengembangan *chatbot* ini adalah menyediakan solusi digital berbasis *Large Language Model (LLM)* yang dapat membantu karyawan dan pihak internal perusahaan dalam memperoleh informasi secara cepat, akurat, dan efisien, tanpa perlu membaca keseluruhan dokumen manual.

Langkah awal dalam pengembangan sistem ini adalah memahami kebutuhan pengguna. Sebelumnya dilakukan diskusi secara langsung dengan supervisor untuk menggali ekspektasi, batasan, serta ruang lingkup dari sistem *chatbot* yang akan dibangun. Hasil diskusi ini menjadi landasan dalam merumuskan strategi teknis, seperti pemilihan model *LLM*, format data yang dibutuhkan, serta skenario penggunaan (*use case*) yang relevan dengan kebutuhan perusahaan.

Dokumen referensi yang digunakan oleh sistem umumnya berformat PDF. Untuk dapat dianalisis oleh model *LLM*, dokumen tersebut perlu dikonversi terlebih dahulu menjadi teks. Proses ini dilakukan menggunakan pustaka *pdfplumber*, yang memungkinkan sistem untuk mengekstrak seluruh isi dokumen dalam bentuk teks.

Dengan format berbasis teks, model *LLM* dapat melakukan proses *information extraction*, seperti pengambilan *keyword*, pemahaman konteks, dan penyusunan respons berbasis isi dokumen.

Gambar 3.19 memperlihatkan proses konversi dokumen ke bentuk teks yang kemudian dijadikan sumber pengetahuan utama bagi *chatbot*.

```python
def upload_pdf():
    file_path = filedialog.askopenfilename(filetypes=[("PDF files", "*.pdf")])
    if not file_path:
        return

    try:
        with open(file_path, "rb") as file:
            pdf_reader = PyPDF2.PdfReader(file)
            text = ""
            for page in pdf_reader.pages:
                text += page.extract_text() + "\n"

        # Split the PDF content into chunks
        chunks = split_into_chunks(text.strip())
        for i, chunk in enumerate(chunks):
            key = f"{os.path.basename(file_path).lower()}_chunk_{i+1}"  # Unique key for each chunk
            save_knowledge_base(key, chunk)

        messagebox.showinfo("Success", "Knowledge base updated successfully from PDF!")
    except Exception as e:
        messagebox.showerror("Error", f"Failed to upload PDF: {e}")
```

Gambar 3.19. *Document to Knowledge*

Tahapan berikutnya adalah mengintegrasikan model LLM ke dalam sistem *chatbot*. Model yang digunakan dalam proyek ini adalah DeepSeek R1-Distill, salah satu model *open-source* dengan ukuran 7 miliar parameter. Untuk memastikan jawaban yang diberikan relevan dan akurat, dilakukan penyusunan *prompt* secara strategis dan kontekstual.

Penyusunan *prompt* menjadi salah satu faktor kunci dalam keberhasilan sistem, karena LLM sangat bergantung pada instruksi yang diberikan. Gambar 3.20 menunjukkan proses pemanggilan model DeepSeek dengan *prompt* yang telah dirancang untuk menyesuaikan dengan konteks pertanyaan pengguna.

```python
model_name = "deepseek-ai/DeepSeek-R1-Distill-Qwen-7B"
print("Loading model and tokenizer...")

try:
    tokenizer = AutoTokenizer.from_pretrained(model_name, padding_side="right")
    tokenizer.pad_token = tokenizer.eos_token  # Explicitly set pad_token to eos_token
    model = AutoModelForCausalLM.from_pretrained(
        model_name,
        torch_dtype=torch.bfloat16,  # Use reduced precision
        device_map="auto",           # Automatically allocate layers to available devices
        offload_folder="offload"     # Enable CPU offloading
    )
    print("Model and tokenizer loaded successfully.")
except Exception as e:
    model = None
    print(f"Error initializing model: {e}")
```

Gambar 3.20. ***DeepSeek Integration***

Setelah dokumen berhasil dianalisis oleh model, seluruh informasi penting yang diperoleh disimpan ke dalam *database* MongoDB. Informasi ini disimpan dalam struktur tertentu sehingga dapat diakses kembali oleh sistem sebagai *knowledge base*.

Dengan adanya *knowledge base* ini, *chatbot* dapat memberikan jawaban yang konsisten dan berbasis data terhadap pertanyaan pengguna. Proses integrasi pengetahuan ke dalam sistem digambarkan pada Gambar 3.21, yang menunjukkan alur penggabungan model LLM, *chatbot*, dan *database* pengetahuan.

```python
def search_knowledge_base(query):
    query = query.lower().strip()
    knowledge_base = load_knowledge_base()  # Reload knowledge base from DB
    matches = difflib.get_close_matches(query, knowledge_base.keys(), n=10, cutoff=0.5)
    responses = [knowledge_base[match] for match in matches]
    return "\n".join(responses) if responses else None
```

Gambar 3.21. ***Bot Knowledge Integration***

Demi meningkatkan kinerja sistem dalam melakukan pencarian informasi, sistem ini mengintegrasikan *ChromaDB*, yaitu *vector database* yang mampu menyimpan pengetahuan dalam bentuk *vector embeddings*. Dengan pendekatan ini, sistem dapat mencari informasi berdasarkan kesamaan makna (*semantic similarity*), bukan hanya berdasarkan kecocokan kata kunci.

Penggunaan *ChromaDB* memungkinkan sistem menjawab pertanyaan pengguna dengan hasil yang lebih relevan secara kontekstual, bahkan ketika pertanyaan menggunakan sinonim atau struktur kalimat yang berbeda. Proses penanganan *user query* berbasis *ChromaDB* ditunjukkan pada Gambar 3.22.

```
def handle_user_query(query):
    knowledge_response = search_knowledge_base(query)
    if knowledge_response:
        return knowledge_response

    # Use the LLM to generate a response if no match is found
    try:
        inputs = tokenizer(query, return_tensors="pt", truncation=True).to(model.device)
        outputs = model.generate(**inputs, max_length=100)
        response = tokenizer.decode(outputs[0], skip_special_tokens=True)
        return response
    except Exception as e:
        print(f"Error generating response: {e}")
        return "Sorry, I couldn't process your query."
```

Gambar 3.22. *Handle User Query*

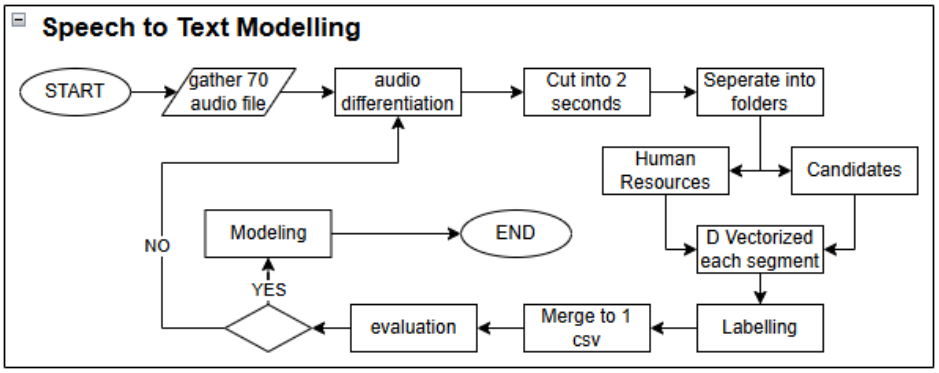
3.2.2.5 Model for Speech to Text

Tugas yang diberikan ini bertujuan untuk membangun sebuah model *machine learning* yang mampu membedakan antara suara dari tim *Human Resource (HR)* dan suara dari kandidat dalam rekaman wawancara. Model ini bertujuan untuk membantu proses pemisahan dan penandaan transkrip berdasarkan pembicara, sehingga analisis wawancara dapat dilakukan dengan lebih terstruktur dan efisien.

Langkah awal dalam pengembangan model adalah memahami permasalahan yang ingin diselesaikan. Langkah awal yang dilakukan merupakan diskusi bersama *supervisor* untuk mendapatkan gambaran menyeluruh terkait kebutuhan pengguna, format data audio yang tersedia, serta tantangan dalam membedakan dua sumber suara yang sering kali memiliki karakteristik akustik yang mirip.

Selain itu, peserta juga mempelajari alur proses wawancara dan bagaimana hasil transkrip akan digunakan oleh tim *HR*. Gambar 3.23

menampilkan *flowchart* pengembangan model *Speech to Text* yang menjelaskan urutan proses dari tahap awal hingga implementasi.

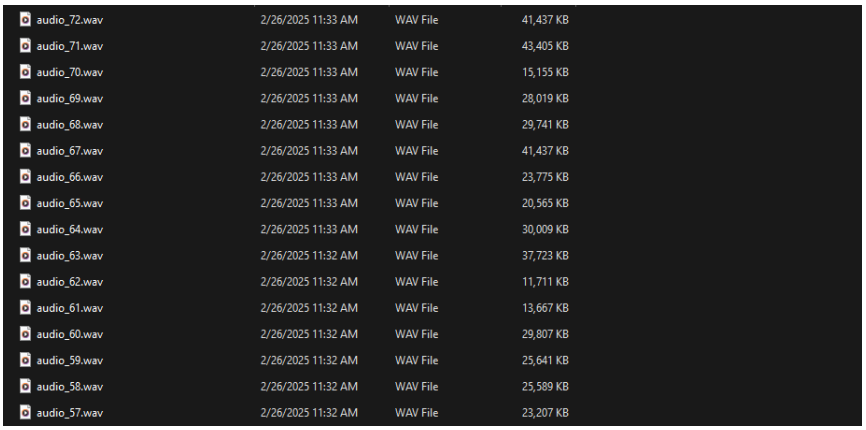


Gambar 3.23. *STT Modeling Flowchart*

Untuk membangun model *machine learning*, peserta mempelajari bagaimana merancang dan menyimpan model dalam format *.h5*, yaitu format standar *HDF5* yang umum digunakan dalam pengembangan model *deep learning*.

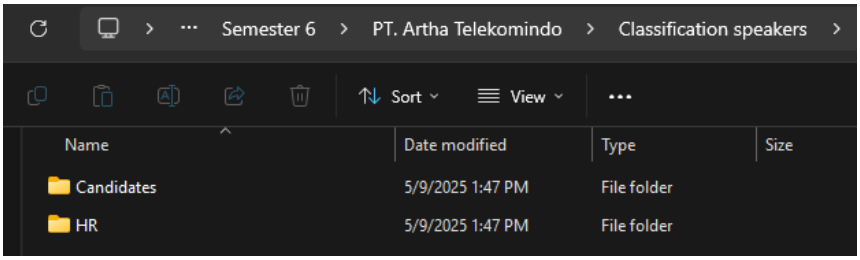
Peserta mengeksplorasi arsitektur model yang sesuai, seperti *Convolutional Neural Network (CNN)* dan *Recurrent Neural Network (RNN)*, yang mampu mengekstraksi pola dari *mel spectrogram*—representasi visual dari audio yang digunakan sebagai fitur dalam proses pelatihan.

Tahap pelatihan model dilakukan secara sistematis. Pertama, peserta mengumpulkan sebanyak 70 file audio hasil rekaman wawancara. Kumpulan data ini ditampilkan pada Gambar 3.24.



Gambar 3.24. ***Dataset***

Setiap file audio dipotong menjadi segmen berdurasi dua detik agar panjang input menjadi seragam. Kemudian, segmen-segmen audio diklasifikasikan ke dalam dua folder berdasarkan sumber suara, yaitu suara dari tim HR dan suara dari kandidat, seperti diperlihatkan pada Gambar 3.25.



Gambar 3.25. ***Classified Audio Files***

Setelah proses segmentasi, dilakukan vektorisasi dengan mengekstraksi fitur audio menggunakan mel spectrogram atau Mel-Frequency Cepstral Coefficients (MFCC). Fitur-fitur ini diubah menjadi

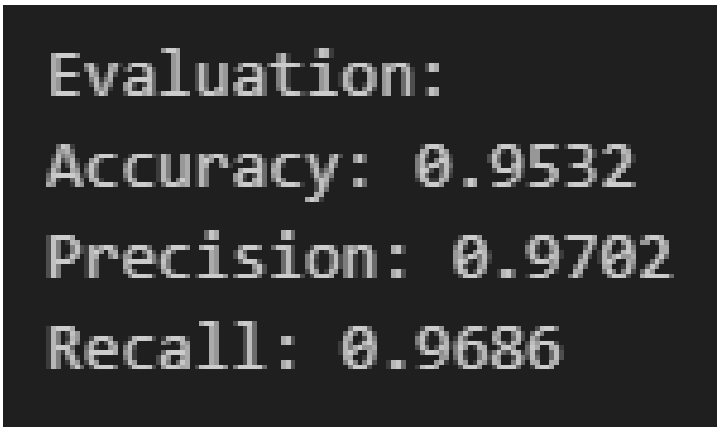
feature vector numerik yang mewakili karakteristik suara. Proses ini digambarkan pada Gambar 3.26.

	mel_1	mel_2	mel_3	mel_4	mel_5	mel_6	mel_7	mel_8	mel_9	mel_10
2	-1.85791	4.257948	10.63688	10.15567	7.056061	5.669002	5.958636	0.954965	1.193372	1.394944
3	0.71554	4.287021	10.63594	9.07421	9.168479	10.1583	7.497809	3.15937	2.614291	1.92389
4	-1.63984	3.957081	9.457941	10.50765	8.938828	7.064003	3.419718	-3.39323	-3.85801	-2.98852
5	-7.54432	-0.03561	4.717633	5.638554	5.256122	6.553371	6.768024	5.409314	4.137487	4.475104
6	-10.9162	-6.59129	-5.49671	-6.9626	0.293563	4.089716	10.7561	9.345372	8.460678	7.776732
7	-6.47752	0.644218	-0.08319	-2.02278	3.667318	3.548216	7.965819	6.567241	8.181827	6.920062
8	-5.0946	1.793222	2.357654	-2.58998	2.858702	-1.95631	3.188437	4.863175	7.027613	6.639903
9	-8.83378	-0.91209	3.576002	-2.04218	5.391368	6.135992	0.044367	9.366431	6.741668	0.093595
10	-7.0054	-3.05493	0.284169	-2.1112	-1.2839	-1.87703	1.049329	2.220853	6.096157	4.844694
11	-7.00649	-0.89398	1.691382	-7.12428	3.502747	0.186182	2.378079	8.843064	3.277342	8.230586
12	-6.67885	-0.27167	-0.38695	-4.40348	0.859556	-4.93861	3.353139	2.912069	7.099913	6.405285
13	-5.65099	-0.30156	-2.24059	-6.8248	-2.60541	-9.6221	0.802149	1.3462	10.88568	12.97842
14	15.78119	23.8037	14.34384	6.067945	3.132665	-0.08983	-7.60536	-11.0699	-11.7306	-16.0578
15	4.901066	10.38125	12.22309	8.142319	4.666822	3.492331	1.90938	0.427896	-2.33456	-7.4774

Gambar 3.26. ***Extracted Log-Mel to Vector***

. Setiap vektor diberi label sesuai dengan sumber suara (HR atau kandidat), kemudian seluruh data dikompilasi dalam satu file .CSV untuk memudahkan proses pelatihan.

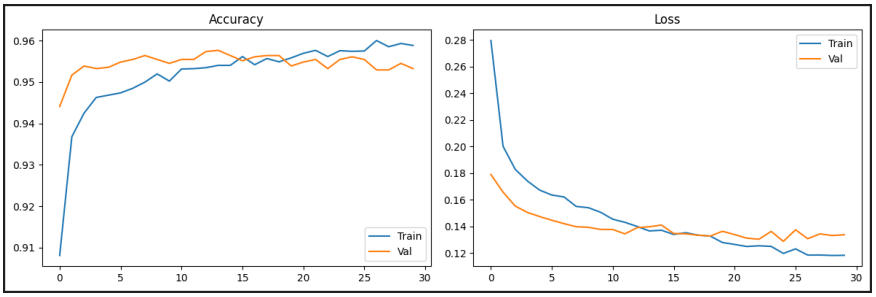
Model klasifikasi dilatih menggunakan data tersebut dengan evaluasi melalui metrik seperti accuracy, precision, recall, dan confusion matrix. Jika hasil belum optimal, maka dilakukan penyesuaian lebih lanjut. Hasil awal dari proses pelatihan model ditampilkan pada Gambar 3.27.



Gambar 3.27. Hasil ***Modeling***

Setelah pelatihan awal, peserta melakukan proses optimasi untuk meningkatkan performa model. Penyesuaian dilakukan pada parameter seperti *learning rate*, *batch size*, dan jumlah *epoch*.

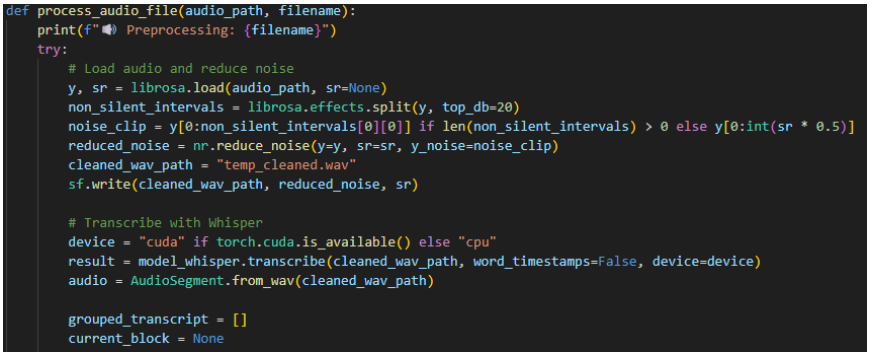
Selain itu, teknik tambahan seperti *data augmentation* dan *feature scaling* juga diterapkan untuk meningkatkan kemampuan generalisasi model. Grafik hasil optimasi, yang menunjukkan kurva *accuracy* dan *loss*, disajikan pada Gambar 3.28.



Gambar 3.28. *Plot Accuracy dan Loss*

Setelah model menunjukkan performa optimal, tahap berikutnya adalah implementasi ke dalam sistem utama. Model digunakan untuk memproses *file* audio wawancara secara otomatis dan mengklasifikasikan setiap segmen audio ke dalam kategori HR atau kandidat.

Hasil klasifikasi ini digunakan untuk menyusun transkrip berdasarkan identitas pembicara, sehingga proses analisis wawancara dapat dilakukan dengan lebih cepat dan sistematis. Implementasi sistem ini ditampilkan pada Gambar 3.29.



```
def process_audio_file(audio_path, filename):
    print(f"🔊 Preprocessing: {filename}")
    try:
        # Load audio and reduce noise
        y, sr = librosa.load(audio_path, sr=None)
        non_silent_intervals = librosa.effects.split(y, top_db=20)
        noise_clip = y[0:non_silent_intervals[0][0]] if len(non_silent_intervals) > 0 else y[0:int(sr * 0.5)]
        reduced_noise = nr.reduce_noise(y=y, sr=sr, y_noise=noise_clip)
        cleaned_wav_path = "temp_cleaned.wav"
        sf.write(cleaned_wav_path, reduced_noise, sr)

        # Transcribe with Whisper
        device = "cuda" if torch.cuda.is_available() else "cpu"
        result = model_whisper.transcribe(cleaned_wav_path, word_timestamps=False, device=device)
        audio = AudioSegment.from_wav(cleaned_wav_path)

        grouped_transcript = []
        current_block = None
```

Gambar 3.29. ***Model Implementation***

3.3 Kendala yang Ditemukan

Selama menjalani masa internship sebagai Data Scientist Intern di PT. Artha Telekomindo, terdapat beberapa kendala yang menjadi tantangan dalam proses adaptasi maupun pelaksanaan tugas yang diberikan. Kendala-kendala tersebut dapat dijabarkan sebagai berikut:

1. **Belum terbiasa dengan istilah teknis dalam perusahaan dan pemrograman.**

   Kurangnya familiaritas terhadap berbagai istilah teknis yang umum digunakan dalam lingkungan perusahaan dan pengembangan sistem berbasis kecerdasan buatan seperti *Retrieval-Augmented Generation (RAG)*, *vectorized*, dan istilah teknis lainnya menjadi salah satu hambatan awal dalam memahami alur kerja yang sedang dikembangkan. Kondisi ini memerlukan waktu adaptasi tambahan untuk mempelajari konsep-konsep baru, baik melalui pembelajaran mandiri maupun melalui bimbingan dari rekan kerja dan supervisor.

2. **Karakteristik perusahaan telekomunikasi yang cepat dan dinamis *(fast-paced)***.

   PT Artha Telekomindo, sebagai perusahaan yang bergerak di bidang teknologi dan telekomunikasi, memiliki karakteristik operasional yang cepat dan dinamis. Lingkungan kerja dengan ritme tinggi ini menuntut

kemampuan adaptasi yang cepat dan efisien terhadap berbagai pembaruan, penugasan baru, serta perubahan kebutuhan proyek yang terjadi secara berkala. Kondisi tersebut menjadi tantangan tersendiri, terutama bagi individu yang belum terbiasa dengan dinamika industri telekomunikasi yang kompetitif.

3. **Kesulitan dalam memahami konteks data dan cara implementasinya**.
   Tantangan selanjutnya berkaitan dengan pemahaman terhadap konteks data yang digunakan, serta proses pengolahan dan implementasinya dalam proyek. Diperlukan waktu tambahan untuk memahami keterkaitan antara struktur data, kebutuhan pengguna (*user requirement*), serta bagaimana data tersebut harus dimodelkan agar relevan dan akurat dalam sistem yang dikembangkan.

4. **Minimnya pengalaman dalam pengembangan *Artificial Intelligence*, khususnya *Large Language Model* (LLM).**
   Keterbatasan pengalaman praktis dalam pengembangan sistem berbasis kecerdasan buatan, khususnya dalam penerapan *Large Language Model* (LLM), menjadi salah satu tantangan dalam proses integrasi LLM ke dalam sistem chatbot berbasis pengetahuan perusahaan yang sedang dikembangkan. Kurangnya pemahaman teknis mengenai kerangka kerja, teknik *fine-tuning*, serta cara kerja LLM memerlukan upaya tambahan untuk mempelajari aspek-aspek tersebut, guna memastikan implementasi sistem dapat berjalan secara optimal.

3.4 Solusi atas Kendala yang Ditemukan

Untuk mengatasi berbagai kendala yang dihadapi selama masa magang, diterapkan sejumlah solusi strategis guna mempercepat proses adaptasi dan meningkatkan efektivitas dalam menyelesaikan tugas. Solusi-solusi berikut dikembangkan dan disesuaikan dengan karakteristik masing-masing kendala:

1. **Solusi atas Ketidakterbiasaan dengan Istilah Teknis dalam Perusahaan dan Pemrograman**

   Dalam menghadapi keterbatasan pemahaman terhadap istilah teknis seperti *Retrieval-Augmented Generation (RAG)*, *vectorized*, dan berbagai terminologi lainnya, dilakukan pendekatan pembelajaran mandiri melalui sumber-sumber terpercaya, seperti dokumentasi resmi, artikel ilmiah, serta video pembelajaran di platform edukatif seperti YouTube dan Coursera. Selain itu, pemahaman teknis juga diperkuat melalui diskusi langsung dengan supervisor dan rekan kerja untuk memperoleh penjelasan praktis sesuai konteks pekerjaan. Pendekatan ini terbukti efektif dalam mempercepat pemahaman terhadap konsep-konsep teknis yang sebelumnya belum dikenal.

2. **Solusi atas Tantangan Bekerja di Lingkungan Perusahaan Telekomunikasi yang Cepat dan Dinamis.**

   Untuk beradaptasi dengan ritme kerja yang cepat dan dinamis, diterapkan strategi manajemen waktu dan prioritas yang lebih terstruktur. Alat bantu seperti *task manager* dan *calendar planning* dimanfaatkan untuk mengatur jadwal harian serta memastikan penyelesaian tugas secara tepat waktu. Di samping itu, penguatan daya tahan mental dan fleksibilitas kerja dilakukan dengan mengadopsi sikap positif terhadap tantangan yang dihadapi, menjadikannya sebagai peluang pembelajaran dalam lingkungan industri yang kompetitif.

3. **Solusi atas Kesulitan Memahami Data dan Implementasinya.**

   Dalam rangka mengatasi hambatan dalam memahami struktur dan konteks data, dilakukan eksplorasi data secara langsung menggunakan visualisasi sederhana dan *statistical summaries* untuk mengidentifikasi pola serta hubungan antar atribut data. Pemahaman terhadap data juga diperkuat dengan mempelajari alur bisnis perusahaan, sehingga konteks penggunaan data dalam proses kerja dapat dipahami secara menyeluruh. Diskusi rutin

bersama tim turut dilakukan untuk memastikan kesesuaian pemahaman terhadap kebutuhan proyek.

4. **Solusi atas Minimnya Pengalaman dalam Pengembangan *Artificial Intelligence*, Khususnya *Large Language Model (LLM)***. Guna menjembatani keterbatasan pengalaman dalam penerapan *Large Language Model*, proses pembelajaran dimulai dari pemahaman dasar mengenai arsitektur *transformer*, konsep *embedding*, serta prinsip *prompt engineering*. Eksperimen langsung dilakukan dengan memanfaatkan model open-source seperti DeepSeek, melalui pengujian berbagai *prompt*, pengaturan *temperature*, dan parameter *max token* untuk mengevaluasi perbedaan hasil keluaran. Dokumentasi resmi dan studi kasus penerapan LLM di industri lain juga dipelajari sebagai referensi tambahan dalam pengembangan sistem chatbot berbasis pengetahuan perusahaan.