# Halaman Judul

*FINE-TUNING* *LARGE LANGUAGE* *MODEL* (LLM) UNTUK MENJAWAB PERTANYAAN DASAR BAGI CALON MAHASISWA BARU DI UNIVERSITAS SYIAH KUALA DENGAN METODE *RETRIEVAL-AUGMENTED GENERATION* (RAG)

TESIS

Diajukan untuk melengkapi tugas-tugas dan

memenuhi syarat-syarat guna pelaksanaan penelitian Tesis

**Oleh:**

HARY RACHMAT

2108207010009



PROGRAM STUDI MAGISTER KECERDASA BUATAN JURUSAN INFORMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS SYIAH KUALA, BANDA ACEH

JULI, 2024

# Halaman Pengesahan Proposal Tesis

PENGESAHAN TESIS

*FINE-TUNING* *LARGE LANGUAGE* *MODEL* (LLM) UNTUK MENJAWAB PERTANYAAN DASAR BAGI CALON MAHASISWA BARU DI UNIVERSITAS SYIAH KUALA DENGAN METODE *RETRIEVAL-AUGMENTED GENERATION* (RAG)

FINE-TUNING LARGE LANGUAGE MODEL (LLM) TO ANSWER BASIC QUESTIONS FOR PROSPECTIVE NEW STUDENTS AT SYIAH KUALA UNIVERSITY USING THE RETRIEVAL-AUGMENTED GENERATION (RAG) METHOD

Oleh:

Nama : Hary Rachmat

NPM : 2108207010009

Program Studi : Kecerdasan Buatan

Menyetujui:

|  |  |
| --- | --- |
| Pembimbing I, | Pembimbing II, |
|  |  |
| Prof. Dr. Ir. Hammam Riza, M.Sc. IPU | Prof. Dr. Taufik Fuadi Abidin, S.Si., M.Tech |
| NIP. 196208081987111001 | NIP. 197010081994031002 |
| Mengetahui: | |
| Ketua Program Studi  Magister Kecerdasan Buatan FMIPA  Universitas Syiah Kuala, | |
|  | |
| Dr. Nizamuddin, M.Info.Sc. | |
| NIP. 197108241996031001 | |

# Pernyataan Bebas Plagiasi

PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan di bawah ini,

Nama lengkap : Hary Rachmat

Tempat/tanggal lahir : Banda Aceh/18 April 1997

NPM : 2108207010009

Program Studi : Kecerdasan Buatan

Jurusan : Informatika

Fakultas : MIPA

Judul Tugas Tesis\* : *Fine-Tuning* *Large Language* *Model* (Llm) Untuk Menjawab Pertanyaan Dasar Bagi Calon Mahasiswa Baru Di Universitas Syiah Kuala Dengan Metode *Retrieval-Augmented Generation* (Rag)

Menyatakan dengan sesungguhnya bahwa Laporan Tesis saya dengan judul seperti di atas adalah **hasil karya saya sendiri** bersama dosen pembimbing dan **bebas plagiasi**.

Jika ternyata dikemudian hari terbukti bahwa Laporan Tesis merupakan hasil plagiasi, saya bersedia menerima sanksi yang berlaku di Universitas Syiah Kuala.

Banda Aceh, 29 Juli 2024

Yang menyatakan,

Hary Rachmat

NPM. 2108207010009

DEKLARASI

(Times New Roman 14, center, bold, capital, dimulai 2 spasi dari margin atas)

(Halaman Deklarasi harus dicantumkan untuk penelitian Tugas Akhir/Tesis yang merupakan bagian dari hibah penelitian dosen atau ide penelitian sepenuhnya berasal dari dosen. Deklarasi dibuat untuk mencegah konflik kepentingan dalam hal kepemilikan Hak Atas Kekayaan Intelektual. Pernyataan dalam Deklarasi dapat mengikuti atau dikembangkan dari contoh di bawah ini)

Penelitian untuk penyelesaian Tesis/Tugas Akhir ini sepenuhnya/sebagian dibiayai oleh hibah penelitian … atas nama …

Data hasil penelitian yang diperoleh merupakan milik …

Banda Aceh, …….….…...

Yang menyatakan,

Nama mahasiswa

NPM.

ABSTRAK

USK Mistral 7B adalah model bahasa besar yang dirancang untuk menjawab pertanyaan dasar penerimaan di Universitas Syiah Kuala (USK). Model ini dibuat menggunakan model open-source Mistral 7B menggunakan data yang dikumpulkan dari penerimaan dan kuliah di universitas. Teknik QLoRA dan RAG digunakan untuk melatih model dan mengambil informasi yang relevan dari sumber data eksternal. Hasilnya dievaluasi menggunakan skor ROUGE. Tanggapan dihasilkan dengan skor >0,5, sepuluh dari 46 pertanyaan dengan metode RAG dan pengujian dengan metode fine-tuning dilakukan pada 20 pertanyaan dan menghasilkan tanggapan dengan skor 1,0 dari semua pertanyaan yang diajukan. Kinerja USK Mistral 7B menunjukkan potensinya sebagai alat yang efektif dalam membantu mahasiswa menanyakan informasi tentang penerimaan dan perkuliahan di USK.

(Times New Roman 12, justified, dimulai 3 spasi dari “ABSTRAK”, maksimal 300 kata)

Kata kunci: Large Language Model, Fine-tuning, Mistral 7B, QLoRA, RAG, ROUGE (4-6 kata kunci)

(Times New Roman 12, justified, dimulai 1 spasi setelah baris terakhir abstrak)

ABSTRACT

(Times New Roman 14, center, bold, capital, dimulai 6 spasi dari kata kunci Abstrak Bahasa Indonesia)

Abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, abstract, dst.

(Times New Roman 12, justified, dimulai 3 spasi dari “ABSTRACT”, maksimal 300 kata)

Keywords: keyword, keyword, keyword, keyword (4-6 keywords)

(Times New Roman 12, justified, dimulai 1 spasi setelah baris terakhir abstract)

Catatan:

1. Abstrak ditulis dalam Bahasa Indonesia dan Bahasa Inggris antara 250-300 kata, menjelaskan secara ringkas tentang penelitian yang dilakukan, metode yang digunakan, hasil signifikan yang didapat dan manfaat dari hasil penelitian.
2. Kata kunci terdiri dari 4-6 kata yang signifikan agar memudahkan pencarian.

# Kata Pengantar

KATA PENGANTAR

Syukur Alhamdulillah dipanjatkan ke hadirat Allah SWT yang telah melimpahkan rahmat-Nya sehingga Tesis yang berjudul “***Fine-Tuning Large Language Model* (LLM) Untuk Menjawab Pertanyaan Dasar Bagi Calon Mahasiswa Baru Di Universitas Syiah Kuala Dengan Metode *Retrieval-Augmented Generation* (RAG)”** dapat diselesaikan.Selawat dan salam disanjungkan kepada Nabi Besar Muhammad SAW.

Tesis ini merupakan salah satu syarat yang harus dipenuhi untuk memperoleh gelar Sarjana/Magister di Jurusan Kimia, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Syiah Kuala. Penyelesaian penulisan Tugas Akhir/Tesis ini tidak terlepas dari bantuan dan dorongan dari berbagai pihak, baik secara moril maupun materil. Pada kesempatan ini, ucapan terimakasih diucapkan kepada:

1. Bapak Dr. Nizamuddin, M.Info.Sc. selaku Koordinator Program Studi Magister Kecerdasan Buatan FMIPA USK.
2. Bapak Prof. Dr. Ir. Hammam Riza, M.Sc. IPU, selaku Pembimbing I dan Bapak Prof. Dr. Taufik Fuadi Abidin, S.Si., M.Tech, selaku pembimbing II yang telah membimbing dan memberikan arahan dan bimbingan kepada penulis dalam penyelesaian Tesis ini.
3. Ayahanda dan Ibunda yang telah mendukung penulis dalam penyelesaian studi di Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Syiah Kuala.
4. Semua pihak yang telah memberikan kontribusi baik moral maupun material sehingga penulis dapat menyelesaikan Proposal Tesis ini.

Semoga tulisan ini dapat bermanfaat bagi perkembangan ilmu pengetahuan.

Banda Aceh, 29 Juli 2024

**Hary Rachmat**

NPM. 2108207010009

# Daftar Isi

DAFTAR ISI

*Halaman*

[Halaman Judul i](#_Toc173234975)

[Halaman Pengesahan Proposal Tesis iii](#_Toc173234976)

[Pernyataan Bebas Plagiasi iv](#_Toc173234977)

[Kata Pengantar vii](#_Toc173234978)

[Daftar Isi viii](#_Toc173234979)

[Daftar Tabel x](#_Toc173234980)

[Daftar Gambar xi](#_Toc173234981)

[Daftar Lampiran xii](#_Toc173234982)

[BAB I PENDAHULUAN 1](#_Toc173234983)

[1.1 Latar Belakang 1](#_Toc173234984)

[1.2 Rumusan Masalah 2](#_Toc173234985)

[1.3 Tujuan Penelitian 3](#_Toc173234986)

[1.4 Manfaat Penelitian 3](#_Toc173234987)

[BAB II TINJAUAN KEPUSTAKAAN 4](#_Toc173234988)

[2.1 *Artificial Intelligence* (AI) 4](#_Toc173234989)

[2.2 *Natural Language Processing* (NLP) 4](#_Toc173234990)

[2.3 *Large Language Models* (LLM) 6](#_Toc173234991)

[2.4 *Chatbot* 9](#_Toc173234992)

[2.5 *Model Mistral 7B* 9](#_Toc173234993)

[2.6 *Model Llama 2* 11](#_Toc173234994)

[2.7 *Fine-tuning* 14](#_Toc173234995)

[2.8 *Retrieval Augmented Generation* (RAG) 15](#_Toc173234996)

[2.9 Kuantisasi 17](#_Toc173234997)

[2.10 GPTQ: *Post-Training Quantization For GPT* 20](#_Toc173234998)

[2.11 GGUF: *GPT-Generated Unified Format* 20](#_Toc173234999)

[2.12 *Dataset* 20](#_Toc173235000)

[2.13 *USK* *Dataset* 22](#_Toc173235001)

[2.14 *Python* 22](#_Toc173235002)

[BAB III METODOLOGI PENELITIAN 24](#_Toc173235003)

[3.1 Tempat dan Waktu Penelitian 24](#_Toc173235004)

[3.2 Alat dan Bahan 24](#_Toc173235005)

[3.3 Metode Penelitian 25](#_Toc173235006)

[BAB IV HASIL DAN PEMBAHASAN 33](#_Toc173235007)

[4.1 Tampilan Chatbot dan Hasil Inferensi LLM 33](#_Toc173235008)

[4.2 Kondisi Dataset 34](#_Toc173235009)

[4.3 Kuantisasi dan Pelatihan Model 35](#_Toc173235010)

[4.4 Pendekatan LLM dengan metode RAG 38](#_Toc173235011)

[4.5 Hasil Pengujian dan Evaluasi Hasil Inferensi 39](#_Toc173235012)

[4.6 Kategori skor ROUGE 39](#_Toc173235013)

[4.7 Menghitung Evaluasi Sumber Daya 40](#_Toc173235014)

[4.8 Analisis Hasil 41](#_Toc173235015)

[BAB V KESIMPULAN DAN SARAN 43](#_Toc173235016)

[5.1 Kesimpulan 43](#_Toc173235017)

[5.2 Saran 44](#_Toc173235018)

[DAFTAR KEPUSTAKAAN 45](#_Toc173235019)

# Daftar Tabel

DAFTAR TABEL

*Halaman*

[Tabel 2.1 Perbandingan Mistral 7B dengan Llama (Sumber: Jiang et al., 2023) 11](#_Toc173235020)

[Tabel 2.2 Distribusi bahasa pada data prapelatihan (Sumber: Touvron et al., 2023) 13](#_Toc173235021)

[Tabel 2.3 Perbadingan performa pada model dasar sumber terbuka 13](#_Toc173235022)

[Tabel 2.4 Perbandingan performa pada model sumber tertutup 13](#_Toc173235023)

[Tabel 2.5 kuantisasi 4-bit menggunakan llama.cpp (Sumber: Neves, 2023) 19](#_Toc173235024)

[Tabel 3.1 Jadwal Kegiatan 24](#_Toc173235025)

[Tabel 3.2 Alat dan Bahan 24](#_Toc173235026)

[Tabel 3.3 Contoh sampel dalam dataset 26](#_Toc173235027)

[Tabel 4.1 Konfigurasi *FastLanguageModel.from\_pretrained* 36](#_Toc173235028)

[Tabel 4.2 Konfigurasi FastLanguageModel.get\_peft\_model 37](#_Toc173235029)

[Tabel 4.3 Konfigurasi *SFTTrainer* 37](#_Toc173235030)

[Tabel 4.4 Konfigurasi *TrainingArguments* 38](#_Toc173235031)

[Tabel 4.5 Nilai skor ROUGE 39](#_Toc173235032)

[Tabel 4.6 Tabel Kategori Nilai Metrik ROUGE (Walker II, 2024) 40](#_Toc173235033)

[Tabel 4.7 Hitungan Waktu pada model saat *fine-tuning* dan menjalankan RAG 40](#_Toc173235034)

# Daftar Gambar

DAFTAR GAMBAR

*Halaman*

[Gambar 2.1 Perbandingan performa pada Model Mistral 7B dengan Llama2 11](#_Toc173235035)

[Gambar 2.2 Alur *Fine-tuning* model (Sumber: Fatyanosa, 2020) 14](#_Toc173235036)

[Gambar 2.3 Arsitektur pada *Retrieval augmented generation* (RAG) 16](#_Toc173235037)

[Gambar 2.4 Tipe data LLM terkuantisasi (Sumber: Neves, 2023) 17](#_Toc173235038)

[Gambar 2.5 Format numerik bfloat16 (Sumber: Neves, 2023) 18](#_Toc173235039)

[Gambar 3.1 *Flowchart* Alur Penelitian 25](#_Toc173235040)

[Gambar 3.2 Alur Pelatihan pada LLM (Sumber: Benveniste, 2023) 27](#_Toc173235041)

[Gambar 3.3 Alur Pada RAG (Sumber: Benveniste, 2023) 29](#_Toc173235042)

[Gambar 3.4 Pipa pengideksan pada RAG (Sumber: Benveniste, 2023) 29](#_Toc173235043)

[Gambar 3.5 Desain UI Chatbot menggunakan Gradio (www.gradio.app, 2024) 32](#_Toc173235044)

[Gambar 4.1 Contoh Hasil inferensi Chatbot menggunakan desain Gradio UI 34](#_Toc173235045)

[Gambar 4.2 Proses *Fine-tuning* pada Model Mistral 7B 35](#_Toc173235046)

[Gambar 4.3 Pipeline RAG, selama fase kueri, mengambil konteks yang paling relevan dari perintah pengguna, meneruskannya ke Model Bahasa Besar 39](#_Toc173235047)

# Daftar Lampiran

DAFTAR LAMPIRAN

|  |  |  |
| --- | --- | --- |
|  |  | *Halaman* |
| Lampiran 1. | Penghitungan data 1 | 21 |
| Lampiran 2. | Penghitungan data 2 | 22 |
| Lampiran 3. | Penghitungan data 3 | 23 |
| Lampiran 4. | Penghitungan data 4 | 24 |
| Lampiran 5. | Penghitungan data 5 | 25 |
| Lampiran 6. | Penghitungan data 6 | 26 |
| Lampiran 7. | Penghitungan data 7 | 27 |
| Lampiran 8. | Biodata Penulis (hanya untuk Laporan TA/Tesis) | 28 |

# BAB I PENDAHULUAN

## Latar Belakang

Dengan kemajuan pesat kecerdasan buatan (AI), model bahasa besar (LLM) telah merevolusi pemrosesan bahasa alami (NLP) sehingga memungkinkan komputer untuk berinteraksi dengan teks dan bahasa. Salah satu contoh LLM adalah *Generative Pretrained Transformer* 4 (GPT-4) yang telah dilatih secara ekstensif pada data teks dalam jumlah besar sehingga memungkinkan GPT-4 melakukan analisis teks di berbagai domain. LLM dapat diaplikasikan pada berbagai bidang seperti bidang kesehatan, pendidikan, hukum, dan bidang-bidang lainnya dengan melakukan pra-pelatihan menggunakan data dalam bidang tersebut dalam jumlah besar.

Salah satu aplikasi chatbot yang populer saat ini adalah ChatGPT yang dikembangkan oleh OpenAI dan dirilis pada akhir tahun 2022 (Mohamadi et al., 2023). Berbagai penelitian berupaya mengeksplorasi kemampuan ChatGPT seperti dalam bidang medis yang dikembangkan untuk membantu dokumentasi klinis (Baker et al., 2024), mengklasifikasikan teks dalam bidang perbankan (Loukas et al., 2023) dan dalam bidang hukum untuk menentukan undang-undang mana yang berpotensi dilanggar (Trozze et al., 2023).

Selain LLM dengan sumber tertutup seperti GPT-4, beberapa penelitian menggunakan LLM sumber terbuka seperti yang dilakukan oleh Huang et al. (2023) dengan mengadaptasi LLM Llama ke domain hukum untuk membantu pengacara dalam membuat laporan teknis. Bhatti et al. (2023) melakukan *fine-tuning* pada LLM Llama 70B yang diberi nama “SM70” untuk menangani berbagai pertanyaan medis serta pengambilan keputusan klinis yang kompleks. Zhao et al. (2023) melakukan *fine-tuning* pada LLM Llama 7B yang diberi nama “Ophtha-LLaMA2” untuk membantu mendiagnosis penyakit mata yang akan memberikan dukungan keputusan bagi dokter. Barandoni et al. (2024) melakukan analisis komparatif LLM untuk mengekstraksi kebutuhan pelanggan travel dari postingan *TripAdvisor* dengan Memanfaatkan beragam model, termasuk model sumber terbuka seperti Mistral 7B dan sumber tertutup seperti GPT-4 dan Gemini. Hasil penelitan menyoroti kemanjuran LLM sumber terbuka, khususnya Mistral 7B, dalam mencapai kinerja yang sebanding dengan model sumber tertutup. Penelitian mengenai chatbot sebagai asisten virtual telah dilakukan oleh Jonatan & Igor (2023) untuk meningkatkan efisiensi layanan kepada pelanggan. Penelitian yang telah dipaparkan tersebut telah menunjukan potensi LLM untuk membantu pekerjaan dalam berbagai bidang.

Berdasarkan data yang di dapat dari website (data.usk.ac.id, 2024) jumlah mahasiswa yang mendaftar ke Universitas Syiah Kuala (USK) mengalami peningkatan setiap tahunnya. Untuk memberikan berbagai informasi seperti detail pendaftaran, uang kuliah, dan lain-lain USK menyediakan website dengan alamat usk.ac.id. sehingga mahasiswa dapat mencari informasi mengenai hal yang berkaitan dengan universitas. Pada saat ada informasi yang belum tersedia di website calon mahasiswa dapat bertanya secara langsung melalui sosial media seperti *direct message* (DM) melalui aplikasi *Instagram*, dan juga menjumpai staf perguruan tinggi secara langsung di bagian Unit Layanan Terpadu (ULT) atau bagian Hubungan Masyarakat (Humas). Penelitian ini akan mengkaji penerapan LLM sebagai asisten virtual dalam bentuk chatbot untuk memberikan informasi interaktif terkait administrasi akademik di USK.

## Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, ada beberapa rumusan masalah yang perlu diteliti untuk menemukan solusinya. Adapun rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Dengan peningkatan jumlah mahasiswa yang mendaftar ke Universitas Syiah Kuala (USK) setiap tahun dibutuhkan alternatif yang dapat membantu staf atau manajemen perguruan tinggi dalam memberikan informasi kepada calon mahasiswa baru.
2. Diperlukan alternatif selain website USK untuk menyediakan informasi mengenai perkuliahan dan detail penerimaan mahasiswa baru di Universitas Syiah Kuala (USK).
3. Bagaimana cara membangun chatbot yang dapat menyediakan informasi mengenai sistem perkuliahan dan detail penerimaan mahasiswa baru di Universitas Syiah Kuala (USK).

## Tujuan Penelitian

Adapun Tujuan dari penelitian ini adalah sebagai berikut:

1. Mengembangkan LLM untuk membantu calon mahasiswa baru mendapatkan informasi di Universitas Syiah Kuala (USK).
2. Menghasilkan LLM baru dengan basis pengetahuan mengenai detail informasi akademik di Universitas Syiah Kuala (USK) serta dapat memberikan jawaban dengan baik dan sesuai.
3. Mengevaluasi efektivitas dan kebermanfaatan terhadap hasil penelitian untuk memastikan bahwa teknologi yang dikembangkan memiliki kinerja yang baik, bermanfaat dalam penggunaan sehari-hari, dan memiliki potensi untuk dikembangkan secara luas dalam skala yang lebih besar.

## Manfaat Penelitian

Beberapa manfaat penelitian yang diharapkan dari penelitianini adalah sebagai berikut:

1. Penyediaan informasi di Universitas Syiah Kuala (USK) yang dapat diakses oleh mahasiswa khususnya bagi calon mahasiswa baru dengan memanfaatkan LLM.
2. Dengan teknologi ini dapat menjadi alternatif bagi calon mahasiswa baru untuk mendapatkan informasi mengenai detail perkuliahan.
3. Peningkatan layanan serta kepuasan yang diberikan kepada mahasiswa di Universitas Syiah Kuala (USK).
4. Menjadi kajian awal dengan pemanfaatan LLM serta mendapatkan ilmu pengetahuan dan informasi baru terkait variasi domain dalam penerapan LLM dalam bidang administrasi akademik di Universitas Syiah Kuala (USK).

# BAB II TINJAUAN KEPUSTAKAAN

## *Artificial Intelligence* (AI)

Menurut Russel, Stuart J; Norvig (2010) Sebuah perangkat pada komputer yang dapat memahami lingkungan di sekitarnya ialah kecerdasan buatan, tujuan dari tindakannya yaitu untuk memberikan suatu respons yang sesuai. Pengertian yang hampir sama juga diberikan oleh Minsky (1989) Menurutnya, kecerdasan buatan pada komputer merupakan suatu ilmu yang mempelajari cara membuat suatu program yang dapat melakukan suatu tindakan yang sama seperti yang manusia dapat lakukan.

Dengan dua pengertian tersebut, maka dapat ditarik kesimpulan bahwa Kecerdasan Buatan (*Artificial Intelligence*) merupakan suatu bidang ilmu yang digunakan dalam hal rancangan, pembangunan dan konsonstruksi pada satu mesin atau komputer pada program komputer hingga dapat dibentuk suatu kecerdasan seperti halnya manusia. Dalam hal ini kecerdasan berarti kemampuan dalam mengambil suatu tindakan, atau penyelesaian suatu masalah seperti halnya manusia yang dapat menggunakan kecerdasannya.

Kecerdasan Buatan (*Artificial Intelligence*) meliputi suatu lingkup penelitian dari berbagai aspek seperti layaknya kecerdasan manusia dalam hal tingkat pengetahuan, pembelajaran, penalaran, perencanaan, pemrosesan bahasa, hingga kemampuan untuk memanipulasi suatu objek. Kecerdasan Buatan (*Artificial Intelligence*) diharapkan dapat memiliki kecerdasan umum layaknya manusia.

## *Natural Language Processing* (NLP)

*Natural language processing* (NLP) adalah salah satu cabang *artificial intelligence* yang membantu komputer memahami bahasa manusia baik secara lisan maupun tertulis, atau yang disebut dengan *natural language*.

Manfaat natural language processing mencakup:

* Mengekstraksi tulisan, di mana NLP mampu merangkum sebuah tulisan dan menemukan data penting dari tulisan tersebut.
* Mendeteksi bahasa pada suatu tulisan lalu menerjemahkannya ke bahasa lain.
* Menghasilkan konten secara otomatis berdasarkan teks yang tidak terstruktur.
* Mendeteksi plagiarisme konten.
* Mengidentifikasi suasana hati atau sentimen dari sebuah tulisan atau suara.
* Mengubah perintah suara menjadi teks tertulis, atau sebaliknnya.

### Jenis-jenis NLP

Dikutip dari OpenGenus, ada 5 model NLP yang paling terbaru dan umum digunakan, yaitu:

* BERT

BERT (*Bidirectional Encoder Representations from Transformers*) dikembangkan oleh Google pada tahun 2018. BERT dirancang untuk membantu komputer memahami makna bahasa ambigu dalam suatu teks dengan cara memanfaatkan kata yang ada di dalamnya untuk membangun konteks.

BERT dapat membaca teks dari dua arah sekaligus dan menganalisis bahasa tulisan menggunakan penyimpanan data pelatihan berlabel khusus.

Data pelatihan berasal dari RNN pra-terlatih yang menggunakan sekumpulan teks tanpa label. Jadi, mesin menggunakan algoritma machine learning untuk menganalisis dan mengelompokkan data tanpa label (unsupervised learning), dan melabeli kalimat atau paragraf sebagai output.

* GPT-3

GPT-3 (*Generative Pre-Trainer* generasi 3) merupakan model terbaru dari GPT dan GPT-2. GPT dibuat oleh OpenAI dan merupakan *machine learning* yang dilatih menggunakan data di dalam internet untuk menghasilkan segala jenis teks.

GPT-3 hanya membutuhkan sedikit input untuk menghasilkan sebuah tulisan yang lengkap.

* XLNet

XLNet bekerja menggunakan metode *autoregresif* (AR), yaitu menggunakan konteks suatu kata untuk memprediksi kata yang akan digunakan setelah atau sebelumnya. Untuk menghasilkan prediksi tersebut, XLNet berfokus pada permutasi atau disebut *Permutation Language Modelling* (PLM).

* ALBERT

ALBERT adalah singkatan dari *A Lite BERT*. ALBERT memiliki desain yang mirip dengan BERT namun parameter yang digunakan jauh lebih sedikit. Misalnya, ALBERT menggunakan SOP (*Sentence Order Prediction*) dan mampu menangkap makna berbeda dari dua kata yang sama, misalnya buah ‘apel’ dan ‘apel’ pagi. Karena hal ini, ALBERT mampu memahami bahasa lebih akurat dibandingkan BERT.

* RoBERTa

RoBERTa dikembangkan dari BERT dengan beberapa modifikasi, seperti menghapus fitur prediksi kalimat setelahnya (*Next Sentence Prediction*) dan dilatih dengan kumpulan kalimat yang lebih panjang (Revou.co, 2024).

## *Large Language Models* (LLM)

Model bahasa besar menggunakan kecerdasan buatan untuk melakukan tugas pemrosesan bahasa alami (NLP) seperti terjemahan, pembuatan teks, menjawab pertanyaan, dan analisis sentimen. LLM menggunakan pembelajaran mendalam (*deep learning*) jenis pembelajaran mesin berbasis AI yang kompleks — untuk memproses data dengan cara yang sangat mirip dengan interaksi alami dan mirip manusia. LLM memungkinkan komputer untuk memahami, memanipulasi, dan menghasilkan teks bahasa manusia. Seiring dengan semakin majunya mereka, banyak tim mulai mengizinkan LLM untuk menyederhanakan dan mengotomatisasi bahkan proses yang berhubungan dengan manusia. LLM tidak hanya dapat menangani tugas rutin yang berulang seperti penjadwalan janji temu atau FAQ, namun mereka juga dapat menangani tugas seperti pengambilan keputusan yang logis dan mengarahkan penelepon dengan tepat ke perwakilan yang tepat. Hal ini karena LLM memiliki akses ke data dalam jumlah besar, sehingga berguna untuk tugas rutin dan lanjutan (Martin, 2023).

### Cara Kerja LLM

LLM dibangun menggunakan apa yang disebut arsitektur berbasis transformator. Transformator adalah jenis arsitektur “jaringan saraf” (seperti otak) yang dirancang oleh Google. Jaringan ini terdiri dari beberapa lapisan, yang semuanya bekerja sama untuk memecah teks menjadi bagian-bagian yang lebih kecil seperti kata atau karakter, yang disebut token, untuk menentukan hubungan dan makna antara setiap token.

Jaringan mencoba mengidentifikasi pola dalam data berurutan (seperti kata-kata yang muncul setelah satu sama lain dalam sebuah kalimat) untuk mengekstrak konteks dan makna dari pola tersebut. Satu lapisan dalam jaringan saraf LLM terdiri dari mekanisme perhatian. Hal ini memungkinkan model untuk fokus pada bagian teks tertentu untuk memahami konteks dan sentimennya. Hal ini membantu model merumuskan respons mirip manusia berdasarkan masukan pengguna. Dapat dibayangkan arsitektur transformator ini seperti otak bayi yang baru lahir ia menerima stimulus dan secara alami mulai mengidentifikasi pola dan membentuk koneksi, oleh karena itu disebut “jaringan saraf” (Martin, 2023).

### Proses Pelatihan LLM

LLM menjalani proses “pelatihan” yang intensif di mana manusia memberi mereka data dalam jumlah besar untuk mengajari mereka cara memahami dan menghasilkan bahasa. Sama seperti bayi yang baru lahir mendengarkan suara orang dewasa dan mencoba memahami semuanya, LLM melakukannya dengan cara yang jauh lebih kompleks. LLM dilatih berdasarkan kumpulan data teks yang sangat besar, seperti buku, artikel, dan bahkan percakapan, namun kemudian harus disesuaikan untuk memberikan hasil yang disesuaikan tergantung pada tugas yang ada. Proses pelatihan LLM melewati dua fase utama: pra-pelatihan (*pretraining*) dan penyesuaian (*fine-tuning*) (Martin, 2023).

### Pra-pelatihan (*pretraining*)

Fase pertama melibatkan pelatihan LLM pada kumpulan data yang besar dan beragam. Jenis data ini dapat diambil dari internet dan dapat mencakup informasi dari media sosial, buku, artikel, dan situs web. Di sinilah model mulai memahami hubungan antara berbagai token (Martin, 2023).

### Penyempurnaan (*fine-tuning*)

Fase berikutnya melibatkan pembelajaran yang diawasi di mana manusia memberikan model dengan kumpulan data tambahan yang memungkinkannya bekerja lebih baik berdasarkan tugas tertentu. Misalnya, LLM dapat dilatih tentang penawaran produk atau petunjuk dan jawaban terkait merek tertentu. Hal ini memungkinkan LLM dapat dengan mudah beradaptasi dengan tugas NLP yang berbeda. Proses pelatihan dua langkah ini memastikan LLM untuk dapat memahami masukan yang berbeda dan menghasilkan informasi yang akurat untuk pengguna (Martin, 2023).

### Pemelajaran ulang untuk kasus penggunaan spesifik

Salah satu manfaat utama dari model bahasa besar adalah fleksibilitas dan adaptabilitasnya. Setelah sebuah model dilatih dengan korpus teks yang besar, model tersebut dapat dilatih ulang untuk kasus penggunaan spesifik, seperti pertanyaan-jawaban atau analisis sentiment. Pemelajaran ulang melibatkan pelatihan model pada dataset yang lebih kecil dan lebih spesifik untuk mengoptimalkan kinerjanya pada tugas tertentu tersebut. Proses ini membutuhkan waktu, namun memungkinkan organisasi tertentu untuk mengembangkan aplikasi pemrosesan bahasa yang sangat akurat dan efektif untuk berbagai kasus penggunaan. Ada banyak alat dan kerangka kerja yang memungkinkan pemelajaran ulang atau penyesuaian model bahasa besar. Banyak penyedia solusi kecerdasan buatan dapat menyesuaikan model bahasa besar yang sudah ada hanya dengan beberapa baris kode dan data pelatihan. Ini memungkinkan mereka menciptakan model bahasa khusus domain atau wilayah yang melebihi performa model bahasa generik untuk kasus penggunaan tertentu. Penyesuaian ini juga memungkinkan personalisasi untuk interaksi dengan pelanggan, meningkatkan pengalaman pengguna (www.wiz.ai, 2023).

### Manfaat pemelajaran ulang pada model bahasa besar

Manfaat pemelajaran ulang yaitu misalkan pada suatu perusahaan yang ingin membangun chatbot untuk tim dukungan pada pelanggan. Perusahaan tersebut ingin chatbot dapat menjawab pertanyaan pelanggan tentang produk dan layanan dengan cepat dan akurat, tetapi juga ingin memiliki sentuhan personalisasi dan mencerminkan gaya dan nada merek perusahaan. Untuk mencapai hal ini, dapat memulai dengan model bahasa besar yang sudah dilatih sebelumnya, seperti GPT-4 yang telah dilatih dengan jumlah data yang besar dan memiliki pemahaman yang kuat terhadap bahasa. Namun, GPT-4 belum dilatih dengan data yang spesifik untuk industri atau produk tertentu, sehingga model tidak memberikan performa yang sesuai dengan yang diharapkan untuk kasus penggunaan ini. Di sinilah *fine-tuning* menjadi penting. Dengan *fine-tuning*, dapat mengambil model yang telah dilatih sebelumnya dan melatihnya menggunakan data pada perusahaan, seperti deskripsi produk, log dukungan pelanggan, dan informasi relevan lainnya. Dalam beberapa kasus, penyesuaian dapat dilakukan dengan menambahkan data baru ke dalam model pelatihan, yang lebih sederhana dan tidak membosankan daripada *fine-tuning*. Apapun metode yang digunakan, hasil akhirnya adalah model belajar tentang informasi mengenai industri dan produk khusus, serta meningkatkan kinerjanya untuk kasus penggunaan tertentu. *Fine-tuning*, bagaimanapun, memberikan lebih banyak personalisasi. Hal ini dapat menggabungkan suara dan nada merek dengan memberikan contoh bagaimana perusahaan dapat berkomunikasi dengan pelanggan. Hal ini membantu memastikan bahwa chatbot terdengar seperti merek dan memberikan pengalaman yang personal untuk pelanggan (www.wiz.ai, 2023).

## *Chatbot*

*Chatbot* adalah sebuah program buatan berbasis *Artificial Intelligence* (AI) atau kecerdasan buatan, yang dapat mensimulasikan percakapan atau obrolan dengan pengguna lain layaknya manusia melalui aplikasi pesan, situs web, aplikasi seluler, atau melalui telepon. teknologi ini juga dikenal sebagai asisten digital yang dapat memahami serta memproses permintaan pengguna, dan memberikan jawaban yang relevan dengan cepat. *Chatbot* sering digambarkan sebagai salah satu ekspresi interaksi paling canggih dan menjanjikan antara manusia dan mesin. Di balik layar, cara kerja *chatbot* terbagi dari dua tugas di inti pertama analisis permintaan pengguna dan memberikan tanggapan. *Chatbot* mengembalikan respons berdasarkan input dari pengguna. Proses ini terlihat sederhana, dalam prakteknya, namun sebenarnya cukup kompleks (Andryanto, 2021).

## *Model Mistral 7B*

Model *chatbot* Mistral 7B merupakan model dengan parameter 7.3B yang dapat digunakan untuk membantu sebagai asisten dalam berbagai tugas pada domain yang spesifik, model *chabot* Mistral 7B ini mudah untuk disesuaikan pada tugas apa pun. Model ini dapat disesuaikan untuk obrolan, yang kinerjanya telah mengungguli model chatbot Llama 2 dengan parameter 13B (Jiang et al., 2023).

Beberapa keunggulan model *chatbot* Mistral 7B dibandingkan model Llama2, yaitu:

1. Mistral 7B mengungguli Llama 2 13B di semua benchmark.
2. Mistral 7B mengungguli Llama 1 34B di banyak tolok ukur
3. Mistral 7B mendekati kinerja CodeLlama 7B pada kode, namun tetap bagus dalam tugas-tugas bahasa Inggris
4. Mistral 7B menggunakan *Grouped-query attention* (GQA) untuk inferensi yang lebih cepat
5. Mistral 7B menggunakan *Sliding Window Attention* (SWA) untuk menangani urutan yang lebih panjang dengan biaya lebih kecil merilis Mistral 7B di bawah lisensi Apache 2.0, dapat digunakan tanpa batasan.

Mistral 7B diuji dengan menjalankan kembali semua tolok ukur dengan jalur evaluasi untuk mengukur kinerja pada berbagai tugas dengan kategori sebagai berikut:

1. **Penalaran Akal Sehat (0-shot):** Hellaswag (Zellers et al., 2020), Winogrande (Sakaguchi et al., 2020), PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), OpenbookQA (Mihaylov et al., 2018), ARC-Easy, ARC-Challenge (P. Clark et al., 2018), CommonsenseQA (Talmor et al., 2019)
2. **Pengetahuan Dunia (5-shot):** NaturalQuestions (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017)
3. **Pemahaman Membaca (0-shot)**: BoolQ (C. Clark et al., 2019), QuAC (Choi et al., 2018)
4. **Matematika**: GSM8K (Cobbe et al., 2021) (8-shot) dengan maj@8 dan Matematika (Hendrycks, Burns, Kadavath, et al., 2021) (4-shot) dengan maj@4
5. **Kode**: Humaneval (Chen et al., 2021) (0-shot) dan MBPP (Austin et al., 2021) (3-shot)
6. **Hasil agregat populer**: MMLU (Hendrycks, Burns, Basart, et al., 2021) (5-shot), BBH (Suzgun et al., 2023) (3-shot), dan AGI Eval (Zhong et al., 2023) (3-5-shot, Hanya pertanyaan pilihan ganda bahasa Inggris)

Semua model dievaluasi ulang pada semua metrik dengan jalur evaluasi untuk perbandingan yang akurat. Mistral 7B secara signifikan mengungguli Llama 2 7B dan Llama 2 13B di semua benchmark. Ia juga jauh lebih unggul dari Llama 1 34B dalam matematika, pembuatan kode, dan tolok ukur penalaran. Performa secara detail model Mistral 7B dengan model Llama 2 di beberapa versi dalam menjalankan semua evaluasi model dapat dilihat pada Gambar 2.1 (Jiang et al., 2023).



Gambar 2.1 Perbandingan performa pada Model Mistral 7B dengan Llama2

(Sumber: Jiang et al., 2023)

Mistral 7B mengungguli Llama 2 13B di semua metrik, dan mendekati performa kode Code-Llama 7B tanpa mengorbankan performa pada tolok ukur non-kode. Tabel hasil evaluasi dapat di lihat pada Tabel 2.1.

Tabel 2.1 Perbandingan Mistral 7B dengan Llama (Sumber: Jiang et al., 2023)



## *Model Llama 2*

Llama 2 adalah kumpulan model bahasa besar (LLM) yang telah dilatih dan disempurnakan dengan skala mulai dari 7 miliar hingga 70 miliar parameter. Llama 2-Chat merupakan LLM yang telah disempurnakan dari beberapa versi model sebelumnya yang dapat digunakan untuk penelitian dan juga merilis model berikut ke masyarakat umum untuk penelitian dan penggunaan komersial penggunaan komersial, Peneliti pada jurnal ini telah merilis beberapa model diantara:

1. Llama 2, versi terbaru dari Llama 1, dilatih dengan menggunakan campuran data baru yang tersedia untuk umum serta meningkatkan ukuran korpus pra-pelatihan sebesar 40%, menggandakan panjang konteks model, dan mengadopsi perhatian kueri yang dikelompokkan (Ainslie et al., 2023). Peneliti pada jurnal ini telah merilis varian Llama 2 dengan parameter 7B, 13B, dan 70B.

2. Llama 2-Chat adalah versi Llama 2 yang disempurnakan dan dioptimalkan untuk kasus penggunaan dialog. Peneliti pada jurnal ini merilis varian model Llama 2-Chat dengan parameter 7B, 13B, dan 70B juga.

Peneliti merangkum kinerja keseluruhan dari serangkaian tolok ukur populer. Tolok ukur tersebut dikelompokkan ke dalam kategori yang tercantum di bawah ini. Hasil untuk seluruh tolok ukur individual tersedia pada

Tabel 2.3 dan Tabel 2.4 (Touvron et al., 2023).

1. **Kode**. Dilaporkan rata-rata pass@1 skor model di HumanEval (Chen et al., 2021) dan MBPP (Austin et al., 2021).
2. **Penalaran Akal Sehat**. Dilaporkan rata-rata PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), HellaSwag (Zellers et al., 2020), WinoGrande (Sakaguchi et al., 2020), ARC Mudah dan Menantang (P. Clark et al., 2018), OpenBookQA (Mihaylov et al., 2018), and CommonsenseQA (Talmor et al., 2019). Dilaporkan 7-shot hasil untuk CommonSenseQA dan hasil 0-shot untuk semua tolok ukur lainnya.
3. **Pengetahuan Dunia**. Dievaluasi kinerja 5-shot di NaturalQuestions (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017) dan melaporkan rata-rata.
4. **Pemahaman Membaca**. Untuk pemahaman bacaan, dilaporkan 0-shot rata-rata di SquAD (Rajpurkar et al., 2018), QuAC, dan BoolQ (C. Clark et al., 2019).s
5. **Matematika**. Dilaporkan rata-rata GSM8K (8 shot) (Cobbe et al., 2021) dan Matematika (4 shot) (Hendrycks, Burns, Kadavath, et al., 2021) tolok ukur di 1 teratas.

Tabel 2.2 dapat dilihat bahwa sebagian besar data prapelatihan dalam bahasa Inggris, artinya Llama 2 akan memiliki performa terbaik untuk kasus penggunaan bahasa Inggris. Kategori besar yang tidak dikenal sebagian terdiri dari data kode pemrograman.

Tabel 2.2 Distribusi bahasa pada data prapelatihan (Sumber: Touvron et al., 2023)



Tabel 2.3 menunjukan performa keseluruhan pada tolok ukur akademik yang dikelompokkan dibandingkan dengan model dasar sumber terbuka. Pada Tabel 2.4 Perbandingan dengan model sumber tertutup pada tolok ukur akademik. Hasil untuk GPT-4 dan GPT-4 berasal dari OpenAI (2023). Hasil untuk model PaLM berasal dari (Chowdhery et al., 2022). Hasil untuk PaLM-2-L berasal dari (Anil et al., 2023).

Tabel 2.3 Perbadingan performa pada model dasar sumber terbuka

(Touvron et al., 2023)



Tabel 2.4 Perbandingan performa pada model sumber tertutup

(Sumber: Touvron et al., 2023)



Model Llama 2 dapat digunakan untuk umum, penelitian dan penggunaan komersial, serta telah dioptimalkan untuk kasus penggunaan dialog. Model Llama 2 mengungguli model obrolan sumber terbuka pada sebagian besar tolok ukur yang telah diuji, dan berdasarkan evaluasi oleh manusia terhadap kegunaan dan keamanan, model Llama 2 dapat menjadi alternatif pengganti yang cocok untuk model sumber tertutup. Peneliti pada jurnal ini memberikan penjelasan rinci tentang pendekatan yang dilakukan terhadap penyempurnaan dan peningkatan kinerja Llama 2-Chat sehingga memungkinkan komunitas mengembangkan model Llama 2 ini dan berkontribusi pada pengembangan LLM yang bertanggung jawab (Touvron et al., 2023).

## *Fine-tuning*

*Fine-tuning* artinya *training model* dengan task yang spesifik menggunakan model yang sudah dilatih dengan dataset yang besar. Biasanya untuk *training language model* menggunakan *unsupervised learning*.



Gambar 2.2 Alur *Fine-tuning* model (Sumber: Fatyanosa, 2020)

Mengapa perlu *fine-tuning*? Pre-trained language model seperti BERT dan GPT telah dilatih menggunakan dataset yang sangat besar. Model yang dihasilkan mampu memahami model bahasa yang dilatih.

Jika tidak menggunakan pre-trained language model, maka diperlukan training dari awal yang pasti membutuhkan banyak waktu dan resource. Jika menggunakan pre-trained language model, maka dengan mudah dapat digunakan model yang sudah dilatih dengan data besar dan tinggal di *fine-tuning* sesuai dengan dataset atau task spesifik. Dengan kata lain, dengan menggunakan *pre-trained language model*, maka dapat lebih cepat mendapatkan hasil yang lebih baik tanpa harus dilatih dari awal.

Membutuhkan data yang lebih sedikit, biasanya dataset untuk *supervised learning* sangat terbatas jumlahnya. Jika *deep learning* dilatih dengan data yang sedikit, ada kemungkinan performanya rendah. Karena *pre-trained language model* sudah dilatih dengan dataset yang besar, maka saat *fine-tuning* hanya perlu dataset yang jumlahnya lebih sedikit.

Memiliki Performa yang lebih baik, sebelum muncul *pre-trained language model*, arsitektur model dibuat sedemikian rupa hingga menghasilkan performa yang lebih tinggi untuk task tertentu. Namun saat ini, model-model tersebut telah dikalahkan oleh *pre-trained language model* seperti *Bidirectional Encoder Representations from Transformers* (BERT) dan *Generative Pre-trained Transformer* (GPT) di berbagai task spesifik NLP (Fatyanosa, 2020).

## *Retrieval Augmented Generation* (RAG)

*Retrieval augmented generation* (RAG) adalah sebuah pendekatan dalam pembuatan model AI yang dapat menghasilkan teks yang menggabungkan kemampuan retrieval (pengambilan informasi) dan generation (pembangkitan teks) (Wicaksono, 2023).

Beberapa fitur utama dari *retrieval augmented generation*:

1. Mengambil informasi relevan dari sumber data eksternal (seperti basis data fakta, artikel, dan lainnya) untuk memperkaya konten yang dihasilkan. Hal ini memungkinkan model untuk menghasilkan teks dengan informasi akurat dan *up-to-date*.

2. Informasi yang diambil kemudian digunakan untuk membimbing proses generate teks. Sehingga output teks yang dihasilkan relevan dengan topik yang diminta dan berisi fakta-fakta akurat.

3. Biasanya selain mengambil informasi faktual, model *retrieval augmented* juga mengambil contoh teks relevan dari dataset. Contoh teks ini kemudian digunakan model sebagai bahan acuan untuk menghasilkan teks baru, sehingga gaya bahasa dan struktur kalimat menjadi lebih natural.

4. Menghasilkan teks dengan kualitas yang lebih baik dan akurat dibandingkan model generatif teks murni, karena didukung oleh komponen *retrieval* di dalam sistemnya.

Jadi secara ringkas, *retrieval augmented generation* adalah penggabungan kemampuan mengambil informasi dan menghasilkan teks sehingga bisa menghasilkan output teks dengan kualitas tinggi. Dengan RAG, model AI tidak hanya mengandalkan pengetahuan yang sudah ada di dalam dirinya, tetapi juga bisa mengambil informasi terbaru dari sumber-sumber eksternal seperti basis data Universitas, kemudian informasi tersebut digunakan untuk membimbing proses generate jawaban.

Manfaat dari penerapan RAG antara lain meningkatkan akurasi, kemutakhiran dan kesesuaian jawaban terhadap konteks pertanyaan pengguna. Selain itu, gaya bahasa jawaban juga bisa disesuaikan agar lebih natural (Wicaksono, 2023). Arsitektur pada *Retrieval augmented generation* (RAG) dapat dilihat pada Gambar 2.3.



Gambar 2.3 Arsitektur pada *Retrieval augmented generation* ([RAG](https://www.researchgate.net/figure/Retrieval-Augmented-Generation-Architecture_fig1_378364457))

(Sumber: Rakotoson, Loïc et al., 2024)

## Kuantisasi

Istilah kuantisasi mengacu pada proses pemetaan nilai tak terhingga yang kontinu ke sekumpulan nilai terhingga yang lebih kecil. Dalam konteks LLM, istilah ini mengacu pada proses konversi bobot model dari tipe data presisi tinggi ke tipe data presisi rendah.

* Presisi Jaringan Syaraf

LLM pada dasarnya adalah jaringan saraf, model komputasi yang direpresentasikan dalam memori GPU atau RAM sebagai Tensor - array angka multidimensi. Untuk menyimpannya, dapat menggunakan berbagai jenis: Float64, Float16, atau bahkan bilangan bulat. Jenis data yang dipilih akan memengaruhi jumlah "digit" yang perlu digunakan dalam memori dan, tentu saja, ukuran memori. Ukuran variabel dapat disebut sebagai presisi, yang menunjukkan berapa banyak "digit" yang digunakan untuk merepresentasikannya dalam memori.



Gambar 2.4 Tipe data LLM terkuantisasi (Sumber: Neves, 2023)

Secara umum, penggunaan presisi tinggi dalam jaringan neural dikaitkan dengan akurasi yang lebih baik dan pelatihan yang lebih stabil. Penggunaan presisi tinggi juga lebih mahal secara komputasi karena memerlukan lebih banyak perangkat keras dan perangkat keras yang lebih mahal. Penelitian yang sebagian besar dilakukan oleh Google dan Nvidia mengenai kemungkinan untuk menggunakan presisi yang lebih rendah untuk beberapa operasi jaringan neural menunjukkan bahwa presisi yang lebih rendah dapat dimanfaatkan untuk beberapa operasi pelatihan dan inferensi.

Selain penelitian, kedua perusahaan mengembangkan perangkat keras dan kerangka kerja untuk mendukung operasi dengan presisi rendah. Misalnya, akselerator Nvidia T4 adalah GPU dengan presisi rendah dan menggunakan teknologi Tensor Cores yang secara signifikan lebih efisien daripada K80. TPU Google memperkenalkan konsep bfloat16, tipe data primitif khusus yang dioptimalkan untuk jaringan saraf. Ide mendasar di balik presisi rendah adalah bahwa jaringan saraf tidak selalu perlu menggunakan semua rentang yang difloat 64bit agar dapat bekerja dengan baik.



Gambar 2.5 Format numerik bfloat16 (Sumber: Neves, 2023)

Seiring dengan semakin besarnya jaringan saraf, pentingnya memanfaatkan presisi yang lebih rendah berdampak signifikan pada kemampuan untuk menggunakannya. Dengan LLM, hal ini menjadi lebih penting lagi. Sebagai referensi, GPU A100 buatan Nvidia memiliki memori 80 GB dalam versi tercanggihnya. Pada Tabel 2.5 di bawah ini, dapat dilihat bahwa model LLama2-70B memerlukan memori sekitar 138 GB, yang berarti bahwa untuk menghostingnya, memerlukan beberapa A100. Mendistribusikan model ke beberapa GPU berarti membayar lebih banyak GPU serta infrastruktur *overhead*. Di sisi lain, versi terkuantisasi memerlukan sekitar 40 GB memori, oleh karena itu dapat dengan mudah masuk ke dalam satu A100, sehingga mengurangi biaya inferensi secara signifikan. Contoh ini bahkan tidak menyebutkan fakta bahwa dalam satu A100, penggunaan model terkuantisasi akan menghasilkan eksekusi yang lebih cepat dari sebagian besar operasi komputasi individual.

Tabel 2.5 kuantisasi 4-bit menggunakan llama.cpp (Sumber: Neves, 2023)

|  |  |  |
| --- | --- | --- |
| Model | Ukuran Asli (FP16) | Ukuran Terkuantisasi (INT4) |
| Llama2-7B | 13,5 GB | 3,9 GB |
| Llama2-13B | 26,1 GB | 7,3 GB |
| Llama2-70B | 138 GB | 40,7 GB |

* Kuantisasi untuk mengurangi ukuran model

Kuantisasi secara signifikan mengurangi ukuran model dengan mengurangi jumlah bit yang diperlukan untuk setiap bobot model. Skenario yang umum adalah pengurangan bobot dari FP16 (Floating-point 16-bit) menjadi INT4 (Integer 4-bit). Hal ini memungkinkan model untuk berjalan pada perangkat keras yang lebih murah dan/atau dengan kecepatan yang lebih tinggi. Dengan mengurangi presisi bobot, kualitas keseluruhan LLM juga dapat mengalami beberapa dampak.

Studi menunjukkan bahwa dampak ini bervariasi tergantung pada teknik yang digunakan dan bahwa model yang lebih besar tidak terlalu terpengaruh oleh perubahan presisi. Model yang lebih besar (lebih dari ~70B) mampu mempertahankan kapasitasnya bahkan saat dikonversi ke 4-bit, dengan beberapa teknik seperti NF4 yang menunjukkan tidak ada dampak pada kinerjanya. Oleh karena itu, 4-bit tampaknya menjadi kompromi terbaik antara kinerja dan ukuran/kecepatan untuk model yang lebih besar ini , sementara 6 atau 8-bit dapat menjadi model yang lebih baik untuk model yang lebih kecil (Neves, 2023).

* Dua Jenis Kuantisasi LLM

Ada dua cara untuk membagi teknik memperoleh model terkuantisasi:

1. ***Post-Training Quantization* (PTQ)**: mengubah bobot model yang sudah dilatih menjadi presisi yang lebih rendah tanpa pelatihan ulang. Meskipun mudah dan sederhana untuk diterapkan, PTQ dapat sedikit menurunkan kinerja model karena hilangnya presisi dalam nilai bobot.
2. ***Quantization-Aware Training* (QAT)**: Tidak seperti PTQ, QAT mengintegrasikan proses konversi bobot selama tahap pelatihan. Hal ini sering kali menghasilkan kinerja model yang unggul, tetapi lebih menuntut komputasi. Teknik QAT yang banyak digunakan adalah QLoRA. (Neves, 2023) (Neves, 2023).

## GPTQ: *Post-Training Quantization For GPT*

GPTQ adalah metode kuantisasi pasca pelatihan (PTQ) untuk kuantisasi 4-bit yang berfokus terutama pada inferensi dan kinerja GPU. Ide di balik metode ini adalah ia akan mencoba mengompresi semua bobot menjadi kuantisasi 4-bit dengan meminimalkan kesalahan kuadrat rata-rata terhadap bobot tersebut. Selama inferensi, ia akan secara dinamis mendekuantisasi bobotnya menjadi float16 untuk meningkatkan kinerja sekaligus menjaga memori tetap rendah (Grootendorst, 2023).

## GGUF: *GPT-Generated Unified Format*

Meskipun GPTQ melakukan kompresi dengan baik, fokusnya pada GPU bisa menjadi kerugian jika tidak memiliki perangkat keras untuk menjalankannya. GGUF, sebelumnya GGML, adalah metode kuantisasi yang memungkinkan pengguna menggunakan CPU untuk menjalankan LLM namun juga memindahkan beberapa lapisannya ke GPU untuk mempercepatnya. Meskipun menggunakan CPU umumnya lebih lambat dibandingkan menggunakan GPU untuk inferensi, ini adalah format yang luar biasa bagi mereka yang menjalankan model pada CPU atau perangkat Apple. Terutama karena dengan melihat pada model yang lebih kecil dan lebih mumpuni yang bermunculan, seperti Mistral 7B, format GGUF akan tetap ada (Grootendorst, 2023).

## *Dataset*

Dataset merupakan kumpulan data yang diatur dalam format yang terstruktur, seperti tabel atau file, dan berisi informasi dari berbagai sumber. Dataset dapat berupa data numerik, teks, gambar, atau gabungan dari semuanya. Keberagaman jenis data set ini memungkinkan aplikasi dalam berbagai industri dan disiplin ilmu.

Dataset dikumpulkan oleh profesional di bidang data, seperti *data analyst*. Untuk bisa digunakan, dataset perlu melewati beberapa tahapan pengolahan data, seperti *data cleaning* dan kategorisasi. Sehingga, dataset yang dapat digunakan biasanya terkumpul berdasarkan kategorinya masing-masing, dan di dalamnya terdapat variabel-variabel yang saling berhubungan (Risyad, 2023).

### Jenis-jenis Dataset

Data dikumpulkan dari berbagai jenis dan berbagai bentuk, sehingga dataset pun dapat dikategorikan dalam beberapa jenis. Seorang data analyst perlulah memahami perbedaan karakter dari jenis-jenis dataset ini karena memiliki fungsi dan tujuan yang berbeda (Risyad, 2023).

1. Numerical Dataset

Numerical dataset adalah jenis dataset yang terdiri dari data numerik saja. Data numerik ini bersifat kuantitatif dan dapat dihitung, diukur, dan diolah menggunakan perhitungan matematis, analisis statistik, atau model prediktif. Contoh data numerik meliputi usia, tinggi badan, suhu, pendapatan, dan banyak lagi.

2. Bivariate Dataset

Bivariate dataset adalah jenis dataset yang berisi dua variabel yang terkait satu sama lain. Dataset jenis ini digunakan untuk menganalisis hubungan atau korelasi antara dua variabel. Contoh bivariate dataset adalah data berat dan tinggi badan dari sekelompok individu, yang setiap entri berisi berat dan tinggi badan masing-masing individu.

3. Multivariate Dataset

Multivariate dataset adalah jenis dataset yang berisi lebih dari dua variabel. Data ini mencakup beberapa atribut atau dimensi, dan setiap entri data memiliki beberapa nilai yang berkaitan dengan atribut atau dimensi tersebut. Analisis untuk jenis dataset ini membantu mengidentifikasi ketergantungan antara banyak variabel. Contoh multivariate dataset adalah data demografi yang mencakup usia, pendapatan, tingkat pendidikan, pekerjaan, dan lain-lain dari sejumlah individu.

4. Categorical Dataset

Categorical dataset adalah jenis dataset yang berisi data dalam bentuk kategori atau kelompok. Data ini bersifat kualitatif dan tidak dapat diukur dengan angka. Contoh data kategorikal meliputi jenis kelamin (laki-laki/perempuan), status pernikahan (menikah/belum menikah/cerai), warna mata (cokelat/biru/hijau), dan sebagainya.

5. Correlation Dataset

Correlation dataset adalah jenis dataset yang berisi nilai-nilai korelasi antara dua atau lebih variabel dalam bentuk matriks atau tabel. Nilai korelasi menggambarkan sejauh mana hubungan linier antara variabel-variabel tersebut. Nilai korelasi berkisar dari -1 hingga +1, dengan nilai -1 menunjukkan korelasi negatif sempurna, 0 menunjukkan tidak adanya korelasi, dan +1 menunjukkan korelasi positif sempurna. Korelasi negatif menandakan bahwa kedua variabel memiliki hubungan yang berbanding terbalik, sedangkan korelasi positif menandakan kedua variabel bergerak ke arah yang sama (Risyad, 2023).

## *USK* *Dataset*

Untuk pelatihan *chatbot* yang diusulkan dalam Penelitian ini, dataset ini terbagi 2 yang pertama sebanyak 20 dataset QnA yang akan digunakan untuk *fine-tuning* dan sebanyak 231 pernyataan diambil dari area "pertanyaan yang sering diajukan oleh mahasiswa baru di USK" di bagian Unit Layanan Terpadu (ULT) Universitas Syiah Kuala (ult.usk.ac.id, 2024) dan disimpan ke dalam format .pdf untuk digunakan dengan metode RAG. Dataset ini secara manual dibagi menjadi beberapa topik berdasarkan topik terkait penerimaan mahasiswa baru di USK. Dataset ini terbagi ke dalam beberapa topik seperti Uang Kuliah (UKTB), Akademik, Umum, beasiswa, dan lain-lain.

## *Python*

Bahasa pemrograman yang bersifat *interpreter* yaitu Python merupakan salah satu bahasa interaktif dengan *object-oriented* beroperasi hampir di semua *platform*, seperti pada Windows, Mac, Linux dan lainnya, dengan sintaksnya yang jelas dan elegan Python dikategorikan sebagai salah satu bahasa pemrograman yang dapat dipelajari dengan mudah, modul - modul pada python dikombinasikan dengan menggunakan struktur data tingkat tinggi yang efisien.

Bahasa pemrograman python sendiri diciptakan di Belanda pada tahun 1990 oleh Guido Van Rossum dan Namanya diambil dari acara televisi kesukaan Guido yaitu Monty Python’s Flying Circus. Secara luas python digunakan sebagai bahasa pemrograman yang dipakai dalam industri dan juga dalam Pendidikan dikarenakan sintaks yang intuitif, ringkas, sederhana dan terdapat pustaka yang banyak.

Python juga merupakan pemrograman yang berorientasi pada objek atau OOP. kelas (*class*) dalam python merupakan sebuah objek berupa data dan merupakan suatu alat yang dapat digunakan dalam pengembangan perangkat lunak yang nantinya dapat digunakan kembali. Python merupakan bahasa yang memiliki kemampuan, menggabungkan kapabilitas, dan sintaksis kode dengan sangat jelas, dan juga dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Walaupun Python tergolong bahasa pemrograman dengan level tinggi, nyatanya Python dirancang sedemikian rupa agar mudah dipelajari dan dipahami. Python memiliki tata bahasa dan script yang sangat mudah untuk dipelajari. Python juga memiliki sistem pengelolaan data dan memori otomatis. Selain itu modul pada Python selalu diupdate (diskominfo.kedirikab.go.id, 2023).

# BAB III METODOLOGI PENELITIAN

## Tempat dan Waktu Penelitian

Penelitian dilakukan di Prodi Magister Kecerdasan Buatan Jurusan Informatika Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Universitas Syiah Kuala. Selama kurang lebih 6 (enam) bulan, peneliti akan melakukan penelitian yang intensif dengan fokus pada topik yang telah ditentukan yang dimulai dari bulan April sampai dengan September 2024. Berikut Tabel 3.1 menampilkan detail jadwal kegiatan penelitian yang akan dilakukan.

Tabel 3.1 Jadwal Kegiatan

| No | Kegiatan | Bulan | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 4 | 5 | 6 | 7 | 8 | 9 |
| 1. | Mengumpulkan dataset berupa informasi penerimaan mahasiswa baru. |  |  |  |  |  |  |
| 2. | Finetuning dataset pada model Mistral 7B. |  |  |  |  |  |  |
| 3. | Melakukan uji coba model Mistral 7B dan dengan metode RAG. |  |  |  |  |  |  |
| 4. | Mengevaluasi model Mistral 7B. |  |  |  |  |  |  |
| 5. | Menyiapkan luaran dan publikasi. |  |  |  |  |  |  |

## Alat dan Bahan

Alat dan bahan yang digunakan pada saat kegiatan penelitian ditunjukkan pada

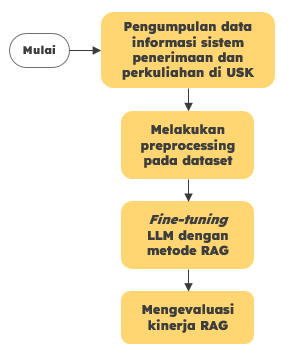
Tabel 3.2 Alat dan Bahan.

Tabel 3.2 Alat dan Bahan

|  |  |
| --- | --- |
| No | Alat dan Bahan |
| 1. | Komputer dengan spesifikasi yang cukup untuk menjalankan Google Colab |
| 2. | Google Colab dengan NVIDIA Tesla T4 GPU |
| 3. | Dataset informasi penerimaan mahasiswa baru di Universitas Syiah Kuala |
| 4. | Model Mistral 7B |

## Metode Penelitian

Penelitian terdiri dari tiga tahapan utama. Tahapan pertama adalah mengumpulkan dataset berupa pertanyaan seputar sistem perkuliahan dan penerimaan mahasiswa baru pada Universitas Syiah Kuala. Metode yang akan digunakan untuk mengumpulkan dataset ini yaitu dengan meminta data pertanyaan terkait sistem perkuliahan dan penerimaan mahasiswa baru ke bagian layanan terpadu USK serta melakukan *preprocessing* pada dataset. Selanjutnya tahap kedua adalah melakukan *fine-tuning* dataset yang sudah terkumpul ke Model Mistral 7B dengan metode RAG dan menggunakan Google Colab dengan GPU T4. Kemudian terakhir pada tahap ketiga menguji Model Mistral 7B dengan mengajukan beberapa pertanyaan dasar mengenai penerimaan mahasiswa baru dan sistem perkuliahan di USK dan mengevaluasi kinerja RAG. Flowchart untuk alur peneliti dapat dilihat pada Gambar 3.1.



Gambar 3.1 *Flowchart* Alur Penelitian

### Pengumpulan dan Persiapan Dataset

Tahap awal adalah mengumpulkan data mentah lebih kurang sekitar 60 sampel dari Unit Layanan Terpadu USK yang mencakup berbagai jenis data mengenai proses administrasi penerimaan dan sistem perkuliahan di USK. Proses pengumpulan data terdiri dari informasi penerimaan mahasiswa baru (misalnya terkait pembayaran UKTB, pengurusan KTM, dan juga Beasiswa). Data tersebut kemudian akan dilakukan *preprocessing* dengan mengubah ke dalam 2 format pertama ke dalam formatinstruksi dengan eksentensi jsonl dan kedua ke dalam format pdf yang akan digunakan untuk mengembangkan LLM dengan domain pertanyaan mengenai USK. Beberapa contoh sampel dari dataset yang digunakan untuk *fine-tuning* pada LLM dapat dilihat pada Tabel 3.3.

Tabel 3.3 Contoh sampel dalam dataset

|  |  |  |
| --- | --- | --- |
| No | Pertanyaan | Jawaban |
| 1 | Bagaimana cara membayar ukt untuk jalur smmptn barat, dan ukt nya apakah di bayar sekalian dengan spi nya? | Salam, silahkan langsung bayar ke bank, dan maslah SPI tidak boleh di cicil kecuali hanya fakultas kedoktean saja |
| 2 | Bagaimana cara mencetak KTM kembali bagi yang hilang? | Bagi mahasiswa lama dapat cetak langsung KTM di laman berkas-akademik.usk.ac.id |
| 3 | Bagaimana cara pembuatan surat rekomendasi kampus untuk beasiswa Bank Indonesia? | Pembuatan surat rekomendasi kampus dapat dibuat di bagian Kemahasiswaan |
| 4 | Saya tidak terdaftar untuk ETD perpustakaan USK | Silahkan untuk melapor/menghubungi pihak Perpustakaan USK. |

### *Fine-tuning* LLM

*Fine-tuning* dilakukan untuk mengkhususkan LLM pada data pelatihan dengan tugas pembelajaran tertentu. Artinya, perlu dilakukan identifikasi terhadap data masukkan, tujuan pembelajaran, dan proses pelatihan dengan benar dan tepat. *Fine-tuning* akan memerlukan alur pelatihan, alur penyajian, dan kemungkinan alur pelatihan berkelanjutan (Benveniste, 2023). Alur pelatihan dapat dilihat pada Gambar 3.2 (Benveniste, 2023).



Gambar 3.2 Alur Pelatihan pada LLM (Sumber: Benveniste, 2023)

* **Registri model**: Registri model akan berisi model asli atau versi berbeda dari model yang disempurnakan. Registri model memerlukan pengambilan versi model dan metadata tambahan yang menjelaskan model tersebut.
* **Modul kuantisasi**: Mengkuantisasi model saat ini merupakan hal yang umum untuk menghemat biaya memori, tetapi ini bukan suatu keharusan. Jika model dikuantisasi, model yang dihasilkan perlu disimpan dalam registri model. Kuantisasi berarti mengubah bobot model dari float menjadi bilangan bulat. Operasi ini dapat membagi ukuran model dengan 4.
* **Penyimpanan fitur**: Seperti halnya pelatihan model ML apa pun, datanya perlu dipersiapkan dengan cermat dan disimpan di penyimpanan fitur. Dalam konteks LLM, persyaratan seputar penyimpanan fitur dapat dilonggarkan jika data pada waktu penayangan hanya dihasilkan oleh pengguna.
* **Modul validasi dan pra-pemrosesan data**: Saat data dimasukkan ke dalam alur pelatihan, data tersebut perlu divalidasi dan kemungkinan besar diproses terlebih dahulu untuk pelatihan.
* **Modifikasi LoRA / QLoRA**: Sekarang sudah umum untuk menyempurnakan LLM menggunakan Adaptor Tingkat Rendah (LoRA) atau versi Terkuantisasinya (QLoRA). Idenya adalah mengganti, dalam model, beberapa matriks besar dengan matriks yang lebih kecil untuk perhitungan gradien. Saat menyempurnakan, hanya memperbarui bobot matriks yang baru disisipkan. Matriks gradien jauh lebih kecil sehingga memerlukan ruang memori GPU yang jauh lebih sedikit. Karena bobot yang telah dilatih sebelumnya dibekukan, sehingga tidak perlu menghitung gradien untuk sebagian besar parameter.
* **Registri LoRA / QLoRA**: Dengan LoRA / QLoRA, diperlukan registri untuk melacak bobot yang telah disesuaikan dan versinya.
* **Modul validasi model**: Seperti pelatihan model lainnya, model yang dihasilkan perlu divalidasi pada data validasi. Ini mengasumsikan dengan memiliki data yang tepat untuk tugas tersebut. Hal ini menjadi rumit karena untuk menyempurnakan model pada tugas tertentu, namun juga dengan mempertahankan kemampuan aslinya. Untuk tugas spesifik, dengan memiliki data validasi yang tepat, dapat kehilangan data yang diperlukan untuk kemampuan aslinya, sehingga tidak dapat dinilai apakah model tersebut tidak melupakan pemrograman sebelumnya (Benveniste, 2023).

### LLM dengan metode RAG

*Retrieval Augmented Generation* atau RAG berarti mengekspos LLM ke data baru yang disimpan dalam database. Tanpa mengubah LLM; sebaliknya, dengan memberikan konteks data tambahan agar LLM dapat menjawab pertanyaan dengan informasi mengenai subjek tertentu. Ide RAG adalah untuk mengkodekan data yang ingin diekspos pada LLM ke dalam embeddings dan mengindeks data tersebut ke dalam database vektor. Saat pengguna mengajukan pertanyaan, pertanyaan tersebut diubah menjadi penyematan, dan dapat digunakan untuk mencari penyematan serupa di dalam database vektor. Setelah menemukan penyematan serupa, kemudian akan dibuatkan prompt dengan data terkait untuk memberikan konteks bagi LLM untuk menjawab pertanyaan tersebut. Kemiripan di sini biasanya diukur menggunakan metrik kesamaan kosinus. Alur pada RAG dapat dilihat pada Gambar 3.3 (Benveniste, 2023).



Gambar 3.3 Alur Pada RAG (Sumber: Benveniste, 2023)

* Sistem Pengindeksan pada RAG

RAG tidak memiliki jalur pelatihan hanya memerlukan pipeline pengindeksan dan pipeline penyajian. Pipa pengindeksan digunakan untuk mengubah data menjadi representasi vektor dan mengindeksnya dalam database Vektor seperti yang dapat dilihat pada Gambar 3.4 (Benveniste, 2023).



Gambar 3.4 Pipa pengideksan pada RAG (Sumber: Benveniste, 2023)

### Evaluasi Kinerja Model

Metrik yang digunakan untuk evaluasi kinerja model pada penelitian ini:

* ***Recall-Oriented Understudy for Gisting Evaluation* (ROUGE)**

Metrik Rouge adalah metrik evaluasi yang umum digunakan dalam tugas NLP untuk membandingkan ringkasan teks yang dihasilkan komputer dengan ringkasan referensi (yang dibuat oleh manusia) (Lin, 2004). Rouge sebagian besar digunakan untuk mengevaluasi tugas peringkasan teks. Nilai metrik Rouges berkisar antara 0 hingga 1. 1 adalah skor yang lebih tinggi, menunjukkan bahwa ringkasan yang dihasilkan komputer dan ringkasan referensi memiliki kesamaan yang tinggi.

Rouge-1, Rouge-2, dan Rouge-L membandingkan dua ringkasan dengan rincian yang berbeda (Amanat, 2024).

* + - 1. Rouge-1

Rouge-1 mengukur ketepatan unigram (kata tunggal) yang tumpang tindih antara teks yang dihasilkan dan teks referensi (buatan manusia).

|  |  |
| --- | --- |
|  | (3.1) |
|  | (3.2) |
|  | (3.3) |

* + - 1. Rouge-2

Rouge-2 mengukur ketepatan bigram yang tumpang tindih antara teks yang dihasilkan dan teks referensi (buatan manusia). Rumus rouge-2 sama dengan rouge-1, namun yang digunakan sepasang kata yaitu bigram, bukan unigram. Bigram mengkompensasi masalah posisi kata Rouge-1 sampai batas tertentu.

* + - 1. Rouge-L

Tidak seperti rouge-1, dan rouge-2, Rouge-L tidak melihat ke dalam unigram atau bigram, melainkan kesesuaian dengan LCS (Longest Common Subsequence) atau urutan kata terpanjang dalam referensi dan teks yang dihasilkan manusia.

|  |  |
| --- | --- |
|  | (3.4) |
|  | (3.5) |
|  | (3.6) |

### Perancangan dan Pengembangan Sistem UI

Pada tahap ini, model yang telah dilatih dengan dataset USK kemudian dilakukan perancangan antarmuka pengguna pada LLM sehingga dapat dihasilkan aplikasi interaktif yang ramah bagi pengguna. Dengan memanfaatkan pustaka Python sumber terbuka yang dapat digunakan untuk membuat demo aplikasi berbasis web. Pustaka Python sumber terbuka yang digunakan, yaitu gradio yang dapat dengan cepat membuat antarmuka pengguna yang indah. Antarmuka pada gradio dapat disesuaikan dengan pengaturan pada judul, deskripsi, dan tampilan. Saat diluncurkan, antarmuka web memungkinkan pengguna untuk berinteraksi dengan chatbot cara kerjanya cukup dengan mengetikkan pertanyaan pada kolom chat dan chatbot akan menghasilkan inferensi jawaban berdasarkan pertanyaan yang diajukan oleh pengguna (Khan, 2023).



Gambar 3.5 Desain UI Chatbot menggunakan Gradio (www.gradio.app, 2024)

# BAB IV HASIL DAN PEMBAHASAN

Luaran yang dihasilkan dalam penelitian ini adalah sebuah aplikasi *chatbot* berbasis web dengan memanfaatkan *Large Language Model* (LLM) *Mistral 7B* yang akan digunakan sebagai alternatif dalam membantu calon mahasiswa baru mendapatkan informasi di Universitas Syiah Kuala (USK). *Chatbot* ini dikembangkan dengan menggunakan kumpulan data USK dengan memanfaatkan kemampuan peringkasan teks LLM untuk menghasilkan teks informasi mengenai sistem perkuliahan dan penerimaan mahasiswa baru di USK. Ada beberapa tahapan yang dilakukan dalam memanfaatkan dan mengembangkan LLM, yaitu dengan mengumpulkan data terkait informasi sistem perkuliahan dan penerimaan mahasiswa baru di USK kemudian kumpulan data tersebut akan dilakukan tahap *preprocesing* dengan mengubah data mentah ke dalam format .csv dan .pdf. Adapun setelah tahap *preprocesing* kemudian dilakukan *finetuning* pada model *Mistral 7B* dan juga untuk memudahkan dalam mengelola data dilakukan *Retrieval Augmented Generation* (RAG) dengan konsep *embeddings*. Tujuan RAG, yaitu untuk mengatasi batasan pada AI generatif karena setiap kali ada pertanyaan yang memerlukan informasi yang berada di luar korpus pelatihan LLM, akan mengakibatkan halusinasi, ketidakakuratan, atau pembelokan pada teks yang dihasilkan. Adapun tahap berikutnya menguji dan mengevaluasi teks yang dihasilkan dengan metode ROUGE untuk membandingkan teks yang dihasilkan oleh model dengan ringkasan referensi yang diberikan. Tahap terakhir dengan membuat UI atau antarmuka web agar dapat diakses oleh pengguna.

## Tampilan Chatbot dan Hasil Inferensi LLM

Dalam penelitian ini *dataset* yang digunakan adalah berupa kumpulan data informasi mengenai sistem perkuliahan dan penerimaan mahasiswa baru di USK dengan jumlah dataset yang didapat sebanyak 231 data pertanyaan dan jawaban. Kumpulan data tersebut akan digunakan pada LLM untuk dapat menghasilkan teks yang mirip dengan teks yang dihasilkan manusia melalui proses yang dikenal sebagai inferensi generatif. Hasil inferensi dari model Mistral 7B dapat dilihat pada Gambar 4.1.



Gambar 4.1 Contoh Hasil inferensi Chatbot menggunakan desain Gradio UI

Seperti yang ditunjukan pada gambar dengan memanfaatkan kemampuan peringkasan teks LLM untuk menghasilkan teks dari masukkan informasi yang telah diberikan chatbot ini tidak hanya sebagai alternatif selain web tetapi juga akan membantu menghemat waktu dan tenaga staf admin universitas.

## Kondisi Dataset

Dataset yang digunakan untuk *fine-tuning* LLM Mistral 7B yaitu sebanyak 20 data terkait informasi mengenai sistem perkuliahan dan penerimaan mahasiswa baru di USK. Dari dataset yang telah didapatkan kemudian dilakukan *preprocessing* data, disini dataset akan diubah kedalam format instruksi Q&A dan disimpan dalam format berekstensi .csv. Dataset ini juga digunakan untuk menjalankan dengan metode RAG yang dibuat dalam bentuk pernyataan dan disimpan ke dalam format berekstensi .pdf. Format dengan ekstensi .csv disimpan ke dalam repositori *huggingface* dan digunakan pada tahap *finetuning* Model. Dan untuk data dengan format .pdf sebanyak 231 data digunakan pada metode RAG dengan tujuan agar data tersebut dapat dikelola dengan lebih mudah.

## Kuantisasi dan Pelatihan Model

Pada Gambar 4.2, proses *fine-tuning* model Mistral 7B dapat dibagi menjadi tiga tahap sebagai berikut:

1. *Fine-tuning* Model dan Persiapan Data

*Fine-tuning*

Mistral 7B

Dataset USK

USK Mistral 7B

Gambar 4.2 Proses *Fine-tuning* pada Model Mistral 7B

Proses *fine-tuning* dilakukan dengan bantuan pustaka Unsloth yang dapat membuat *fine-tuning* LLM 5x lebih cepat dan menggunakan VRAM 80% lebih sedikit. Model hasil *fine-tuning* selanjutnya dilakukan kuantisai ke dalam format 4bit dengan teknik QLoRA. Tujuan kuantisai adalah untuk mengurangi presisi *floating-point* bobot model melalui proses kuantisasi ini model akan dikonversi dari presisi tinggi ke presisi rendah. Dalam proses ini, bobot model di konversi ke format int4 melalui lapisan kuantisasi dan disimpan di GPU. Komputasi inti dilakukan pada CUDA, yang dapat mengurangi konsumsi memori dan meningkatkan efisiensi. Hal ini memungkinkan untuk menyempurnakan model yang lebih besar pada GPU tingkat konsumen. Model hasil *fine-tuning* disimpan dalam repositori *huggingface* dalam format GGUF*.*

1. *Pre-trained* Model

Sebelum tahap pelatihan, dilakukan analisis terhadap LLM terkenal dari kumpulan LLM yang ada saat ini untuk memilih model yang paling sesuai untuk tujuan penelitian ini. Evaluasi pada LLM terutama dengan mempertimbangkan beberapa aspek utama seperti kemampuan beradaptasi domain, kompatibilitas dengan standar di bidang akademik, kemampuan bilingual, ketersediaan sumber terbuka, efisiensi parameter, serta pertimbangan biaya dan perizinan. Pertimbangan ini juga melihat beberapa metrik evaluasi yang telah diuji di berbagai tolok ukur.

1. *Parameter-Efficient Fine-Tuning* (PEFT)

Berdasarkan pertimbangan kumpulan data yang terbatas dan untuk mengurangi biaya pelatihan serta memitigasi potensi risiko, peneliti memilih metode *Progressive Layer Freezing and Fine-tuning* (PEFT) untuk menyempurnakan model Mistral 7B. Metode PEFT saat ini merupakan salah satu pendekatan utama untuk menyempurnakan LLM. PEFT secara selektif menyempurnakan sejumlah kecil parameter model tambahan, sehingga secara signifikan dapat mengurangi biaya komputasi dan penyimpanan, serta memungkinkan adaptasi yang efisien dari LLM yang telah dilatih sebelumnya ke berbagai domain aplikasi.

Pada penelitian ini Secara khusus menggunakan metode penyesuaian tingkat rendah (LoRA) (Hu et al., 2022) untuk menyempurnakan model bahasa besar. LoRA melibatkan pembekuan bobot pra-pelatihan dan pembuatan matriks versi peringkat rendah untuk lapisan kueri dan nilai. Matriks peringkat rendah ini memiliki parameter yang jauh lebih sedikit dibandingkan model aslinya, sehingga memungkinkan penyesuaian dengan penggunaan memori GPU yang lebih sedikit.

Pengaturan *hyperparameter* yang digunakan pada proses *fine-tuning* menggunakan metode *QLoRa* ditunjukkan pada Tabel. Pada Tabel 4.1 menunjukkan konfigurasi dalam membuat instance *FastLanguageModel.from\_pretrained*, menggunakan komponen *unsloth*, dengan konfigurasi spesifik seperti panjang urutan maksimum, tipe data, dan pemuatan 4-bit (V., 2024).

Tabel 4.1 Konfigurasi *FastLanguageModel.from\_pretrained*

|  |  |
| --- | --- |
| **Parameter** | **Nilai** |
| model\_name | mistralai/Mistral-7B-v0.1 |
| max\_seq\_length | 2048 |
| dtype | None |
| load\_in\_4bit | True |

Pada Tabel 4.2 Objek *FastLanguageModel* menyediakan atribut *get\_peft\_model* tempat untuk dapat mengkonfigurasi berbagai parameter untuk penyesuaian, seperti jumlah *attention heads*, modul target, *dropout rate*, *LoRa alpha*, dan banyak lagi. Penggunaan *gradien* *checkpointing* dan teknik canggih lainnya menunjukkan kemampuan *unsloth* untuk mengoptimalkan kinerja model (V., 2024).

Tabel 4.2 Konfigurasi FastLanguageModel.get\_peft\_model

|  |  |
| --- | --- |
| **Parameter** | **Nilai** |
| r | 16 |
| target\_modules | ["q\_proj", "k\_proj", "v\_proj", "o\_proj", "gate\_proj", "up\_proj", "down\_proj",], |
| lora\_alpha | 16 |
| lora\_dropout | 0 |
| bias | None |
| use\_gradient\_checkpointing | "unsloth" |
| random\_state | 3407 |
| use\_rslora | False |
| loftq\_config | None |

Langkah berikutnya pada Tabel 4.3 dan Tabel 4.4 adalah menginisialisasi *Supervised Fine-tuning Trainer* yang membantu proses fine-tuning. Dimulai dengan menginisialisasi model, bersama dengan kumpulan data yang harus disempurnakan, tokenizer dan semua Argumen Pelatihan yang diperlukan (kecepatan pembelajaran, langkah maksimum, penurunan bobot, pengoptimalan, dll) (V., 2024).

Tabel 4.3 Konfigurasi *SFTTrainer*

|  |  |
| --- | --- |
| **Parameter** | **Nilai** |
| model | model |
| tokenizer | tokenizer |
| train\_dataset | dataset |
| dataset\_text\_field | "text" |
| max\_seq\_length | 2048 |
| dataset\_num\_proc | 2 |
| packing | False |

Tabel 4.4 Konfigurasi *TrainingArguments*

|  |  |
| --- | --- |
| **Parameter** | **Nilai** |
| per\_device\_train\_batch\_size | 2 |
| gradient\_accumulation\_steps | 4 |
| warmup\_steps | 5 |
| max\_steps | 60 |
| learning\_rate | 2e-4 |
| fp16 | not is\_bfloat16\_supported() |
| bf16 | is\_bfloat16\_supported() |
| logging\_steps | 1 |
| optim | "adamw\_8bit" |
| weight\_decay | 0.01 |
| lr\_scheduler\_type | "linear" |
| seed | 3407 |
| output\_dir | "outputs" |

## Pendekatan LLM dengan metode RAG

Pada tahap ini dilakukan pendekatan pada LLM dengan metode RAG, pendekatan dengan metode ini bertujuan untuk mengatasi batasan pada AI generatif ketika memerlukan informasi yang berada di luar korpus pelatihan LLM, sehingga dengan metode ini akan menghindari LLM yang akan menghasilkan teks yang tidak akurat, halusinasi, atau pembelokan pada saat memberikan jawaban dari pertanyaan yang diberikan. pada metode RAG data yang digunakan yaitu dokumen eksternal yang berisi informasi sistem akademik USK dan di simpan dalam format berekstensi .pdf. Kumpulan data informasi tersebut kemudian dilakukan embedding untuk mengubah teks menjadi vektor sehingga dapat disimpan ke dalam vektor database. Vektor database yang digunakan pada penelitian ini yaitu FAISS. Model yang akan digunakan pada tahap ini yaitu model yang telah terkuantisasi ke dalam format *GPT-Generated Unified Format* (GGUF) sehingga memungkinkan untuk menggunakan CPU saat menjalankan LLM dengan memindahkan beberapa lapisannya ke GPU sehingga dapat mempercepat kinerja model dalam menghasilkan teks. Proses kueri pada model dalam pengambilan konteks yang paling relevan dari perintah pengguna dengan metode RAG ditampilkan pada Gambar 4.3



Gambar 4.3 Pipeline RAG, selama fase kueri, mengambil konteks yang paling relevan dari perintah pengguna, meneruskannya ke Model Bahasa Besar

(Sumber: docs.llamaindex.ai, 2023)

## Hasil Pengujian dan Evaluasi Hasil Inferensi

Pada tahap ini dilakukan pengujian chatbot dengan memberikan pertanyaan, hasil inferensi teks yang dihasilkan kemudian dihitung dengan menggunakan metode *Recall-Oriented Understudy for Gisting Evaluation (*ROUGE) skor. Dengan menggunakan metrik ROUGE, tanggapan dari chatbot dapat dievaluasi untuk melihat kesamaan antara referensi secara kuantitatif dan jawaban yang dihasilkan oleh chatbot. Hasil perhitungan ROUGE skor pada model ditunjukan pada Tabel 4.5.

Tabel 4.5 Nilai skor ROUGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metode | Jumlah Pertanyaan | Skor ROUGE | | |
| R-1 | R-2 | R-L |
| *Fine-tuning* | 20/20 | 1.0 | 1.0 | 1.0 |
| RAG | 15/56 | >0.5 | >0.5 | >0.5 |

## Kategori skor ROUGE

Skor ROUGE yang baik bervariasi berdasarkan tugas pada ringkasan dan metrik. Skor ROUGE-1 dengan kategori sangat baik dengan nilai sekitar 0,5, dan skor di atas 0,5 dianggap baik dan 0,4 hingga 0,5 sedang. Untuk ROUGE-2, skor di atas 0,4 masuk ke dalam kategori baik, dan 0,2 hingga 0,4 masuk ke dalam kategori sedang.

Skor ROUGE-L dengan kategori bagus mendapatkan nilai sekitar 0,4 dan kategor rendah berkisar pada nilai 0,3 hingga 0,4. Meskipun skor ROUGE berguna, skor tersebut tidak memperhitungkan kualitas semantik atau sintaksis dan harus dilengkapi dengan metrik lain dan evaluasi manusia untuk penilaian yang lengkap (Walker II, 2024).

Tabel 4.6 Tabel Kategori Nilai Metrik ROUGE (Walker II, 2024)

|  |  |  |  |
| --- | --- | --- | --- |
| Metrik ROUGE | Baik sekali | Baik | Sedang |
| ROUGE-1 | 0.5+ | >0.5 | 0.4-0.5 |
| ROUGE-2 | - | >0.4 | 0.2-0.4 |
| ROUGE-L | - | ~0.4 | 0.3-0.4 |

## Menghitung Evaluasi Sumber Daya

Pada penelitian ini peneliti melakukan uji waktu pada model Mistral 7B, dengan pengujian *fine-tuning* dan inferensi. Peneliti menggunakan satu kartu grafis NVIDIA Tesla T4 GPU yang tersedia di Google Colab untuk inferensi dan juga untuk *fine-tuning*. Berdasarkan uji waktu *fine-tuning* dan menggunakan 20 dataset, seperti yang ditunjukkan pada Tabel 4.7, Peneliti menyimpulkan bahwa pendekatan *fine-tuning* dengan library *unsloth* efektif dan efisien, serta membutuhkan waktu yang minimal. Pengujian waktu inferensi menunjukkan bahwa USK Mistral 7B tidak memerlukan daya komputasi yang berlebihan selama proses inferensi, sehingga menjadikannya sistem hemat energi dan berefisiensi untuk merespons pertanyaan dari pengguna hanya saja membutuhkan waktu 5-6 menit untuk merespons pertanyaan pengguna. Harapannya model ini akan menjadi sistem yang efisien dalam menangani informasi terkait administrasi akademik USK di masa mendatang.

Tabel 4.7 Hitungan Waktu pada model saat *fine-tuning* dan menjalankan RAG

|  |  |  |
| --- | --- | --- |
| Model | Waktu *Fine-tuning* (jam) | Waktu RAG (menit) |
| USK Mistral 7B | 2 | 5-6 |

## Analisis Hasil

* Masalah Data Pelatihan

Faktor penting yang berkontribusi terhadap halusinasi LLM adalah sifat data pelatihan. LLM, seperti Mistral 7B, GPT, Falcon, dan Llama, menjalani pelatihan ekstensif tanpa pengawasan dengan kumpulan data yang besar dan beragam dari berbagai asal. Memverifikasi keadilan, ketidakberpihakan, dan kebenaran faktual data ini merupakan sebuah tantangan. Saat model ini belajar menghasilkan teks, model tersebut juga dapat menemukan dan mereplikasi ketidakakuratan faktual dalam data pelatihan. Hal ini mengarah pada skenario dimana model tidak dapat membedakan antara kebenaran dan fiksi dan dapat menghasilkan keluaran yang menyimpang dari fakta atau penalaran logis.LLM yang dilatih berdasarkan kumpulan data yang bersumber dari internet dapat berisi informasi yang bias atau salah. Informasi yang salah ini dapat menyebar ke keluaran model, karena model tidak dapat membedakan antara data yang akurat dan tidak akurat.

* Mengurangi Halusinasi

Upaya untuk mengurangi halusinasi sangat penting untuk menjaga kredibilitas dan fungsionalitas LLM. Metode utama untuk mengidentifikasi dan mengurangi kesalahan ini melibatkan kombinasi metrik canggih dan evaluasi kritis manusia. Ini termasuk:

* Metrik kualitas linguistik seperti ROUGE dan BLEU
* Metrik validitas konten, yaitu berbasis IE, berbasis QA, dan berbasis NLI
* *FactScore* untuk memeriksa keakuratan fakta individu
* Metode *Retrieval-Augmented Generation* (RAG)

Metode inovatif seperti *SelfCheckGPT* mendeteksi halusinasi dengan menilai konsistensi beberapa jawaban yang dihasilkan untuk pertanyaan yang sama. Selain itu, teknik seperti *chain-of-thought prompting* dan *Retrieval-Augmented Generation* (RAG) terus dieksplorasi untuk memperkuat kemampuan model dalam memberikan informasi yang tepat dan relevan.

* Pengaruh GPU dalam Implementasi LLM

Dalam menjalankan LLM, GPU memegang peranan penting. GPU khusus dengan VRAM tinggi dapat mempercepat komputasi yang dibutuhkan oleh model secara signifikan. Pada penelitian ini GPU yang digunakan yaitu “NVIDIA Tesla T4 GPU” yang tersedia di Google Colab secara gratis, hasil pengujian dengan GPU ini menggunakan metode RAG membutuhkan waktu 4-5 menit untuk dapat menghasilkan respons dari pertanyaan yang diajukan.

# BAB V KESIMPULAN DAN SARAN

## Kesimpulan

*Fine-tuning* pada model Mistral 7B menjadi USK Mistral 7B memerlukan banyak data sehingga dapat menghasilkan LLM yang lebih baik dalam menjawab pertanyaan terkait sistem perkuliahan dan penerimaan mahasiswa baru di USK. Dan membutuhkan waktu yang lama lebih kurang 2 jam untuk mendapatkan model hasil *fine-tuning* dengan dataset sebanyak 20 data pertanyaan dan jawaban.

Metode *Retrieval Augmented Generation* (RAG) mengatasi batasan pada AI generatif ketika memerlukan informasi yang berada di luar korpus pelatihan LLM, sehingga dengan metode ini akan menghindari LLM yang akan menghasilkan teks yang tidak akurat, halusinasi, atau pembelokan pada saat memberikan jawaban dari pertanyaan yang diberikan. Metode RAG ini dapat digunakan untuk menghasilkan jawaban dengan lebih cepat karena menggunakan data eksternal. Metode RAG memungkinkan model untuk menghindari batasan pada model generative AI. Respon yang dihasilkan dengan metode RAG mampu menghasilkan jawaban yang cukup baik dengan melihat dari nilai ROUGE skor yang telah di uji.

Berdasarkan metrik teknik dan penilaian sumber daya komputasi yang *open-source*, hal ini dapat disimpulkan bahwa USK Mistral 7B memiliki potensi untuk diterapkan karena dengan konsumsi energi yang rendah dapat menghasilkan respons yang memiliki skor ROUGE >5.

LLM, khususnya Mistral 7B memiliki potensi yang luar biasa dalam penerapannya di berbagai bidang seperti di bidang pelayanan akademik dan administrasi. Pada penelitian ini menunjukkan bahwa dengan pelatihan sejumlah kecil data informasi terkait akademik dan administrasi di USK, USK Mistral 7B mampu merespons berbagai pertanyaan dengan baik. Kinerja LLM yang mengesankan seperti Mistral 7B menyoroti kemampuannya sebagai suatu alat yang ampuh untuk membantu Mahasiswa dalam mendapatkan informasi di USK.

## Saran

Disarankan untuk penelitian selanjutnya selain menambah dataset agar model menjadi lebih handal diperlukan pengujian pada model-model lain yang lebih baru dengan fitur-fitur yang lebih canggih, dengan menjelajahi hal tersebut dapat menemukan perbandingan pada model-model lain sehingga mendapat model yang terbaik untuk domain pelayanan administrasi akademik.

# DAFTAR KEPUSTAKAAN

Ainslie, J., Lee-Thorp, J., de Jong, M., Zemlyanskiy, Y., Lebrón, F., & Sanghai, S. (2023). GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints. *EMNLP 2023 - 2023 Conference on Empirical Methods in Natural Language Processing, Proceedings*, 4895–4901. https://doi.org/10.18653/v1/2023.emnlp-main.298

Amanat, M. U. (2024). *LLM evaluation with Rouge*. Medium.Com. https://medium.com/@MUmarAmanat/llm-evaluation-with-rouge-0ebf6cf2aed4

Andryanto, S. D. (2021). *Apa itu Chatbot? Begini Cara Kerja Asisten Digital ini*. Tekno.Tempo.Co. https://tekno.tempo.co/read/1536593/apa-itu-chatbot-begini-cara-kerja-asisten-digital-ini

Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., Chu, E., Clark, J. H., Shafey, L. El, Huang, Y., Meier-Hellstern, K., Mishra, G., Moreira, E., Omernick, M., Robinson, K., … Wu, Y. (2023). *PaLM 2 Technical Report*. *May*. http://arxiv.org/abs/2305.10403

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., Jiang, E., Cai, C., Terry, M., Le, Q., & Sutton, C. (2021). *Program Synthesis with Large Language Models*. 1–34. http://arxiv.org/abs/2108.07732

Baker, H. P., Dwyer, E., Kalidoss, S., Hynes, K., Wolf, J., & Strelzow, J. A. (2024). ChatGPT’s Ability to Assist with Clinical Documentation: A Randomized Controlled Trial. *Journal of the American Academy of Orthopaedic Surgeons*, *32*(3), 123–129. https://doi.org/10.5435/JAAOS-D-23-00474

Barandoni, S., Chiarello, F., Cascone, L., & Puccio, S. (2024). *Automating Customer Needs Analysis : A Comparative Study of Large Language Models in the Travel Industry*.

Benveniste, D. (2023). *Augmenting LLMs: Fine-Tuning or RAG?* Newsletter.Theaiedge.Io. https://newsletter.theaiedge.io/p/augmenting-llms-fine-tuning-or-rag

Bhatti, A., Parmar, S., & Lee, S. (2023). *SM70: A Large Language Model for Medical Devices*. *1*, 1–5.

Bisk, Y., Zellers, R., Le Bras, R., Gao, J., & Choi, Y. (2020). PIQA: Reasoning about physical commonsense in natural language. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, 7432–7439. https://doi.org/10.1609/aaai.v34i05.6239

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. de O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., Ray, A., Puri, R., Krueger, G., Petrov, M., Khlaaf, H., Sastry, G., Mishkin, P., Chan, B., Gray, S., … Zaremba, W. (2021). *Evaluating Large Language Models Trained on Code*. http://arxiv.org/abs/2107.03374

Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W. T., Choi, Y., Liang, P., & Zettlemoyer, L. (2018). QUAC: Question answering in context. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, 2174–2184. https://doi.org/10.18653/v1/d18-1241

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., … Fiedel, N. (2022). *PaLM: Scaling Language Modeling with Pathways*. 1–87. http://arxiv.org/abs/2204.02311

Clark, C., Lee, K., Chang, M. W., Kwiatkowski, T., Collins, M., & Toutanova, K. (2019). Boolq: Exploring the surprising difficulty of natural yes/no questions. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *1*, 2924–2936.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., & Tafjord, O. (2018). *Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge*. http://arxiv.org/abs/1803.05457

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., & Schulman, J. (2021). *Training Verifiers to Solve Math Word Problems*. 1–22. http://arxiv.org/abs/2110.14168

data.usk.ac.id. (2024). *Data Mahasiswa Daftar Ulang*. Data.Usk.Ac.Id. https://data.usk.ac.id/mahasiswa-daftar

diskominfo.kedirikab.go.id. (2023). *Apa itu bahasa Python?* Diskominfo.Kedirikab.Go.Id. https://diskominfo.kedirikab.go.id/baca/apa-itu-bahasa-python

docs.llamaindex.ai. (2023). *High-Level Concepts*. Docs.Llamaindex.Ai.

Fatyanosa, T. (2020). *Fine-Tuning Pre-Trained Transformer-based Language Model*. Medium.Com. https://fatyanosa.medium.com/fine-tuning-pre-trained-transformer-based-language-model-c542af0e7fc1

Grootendorst, M. (2023). *Which Quantization Method is Right for You? (GPTQ vs. GGUF vs. AWQ)*. Maartengrootendorst.Com. https://www.maartengrootendorst.com/blog/quantization/

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., & Steinhardt, J. (2021). Measuring Massive Multitask Language Understanding. *ICLR 2021 - 9th International Conference on Learning Representations*.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., & Steinhardt, J. (2021). *Measuring Mathematical Problem Solving With the MATH Dataset*. *NeurIPS*, 1–22. http://arxiv.org/abs/2103.03874

Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). Lora: Low-Rank Adaptation of Large Language Models. *ICLR 2022 - 10th International Conference on Learning Representations*, 1–26.

Huang, Q., Tao, M., Zhang, C., & An, Z. (2023). *Lawyer LLaMA: Enhancing LLMs with Legal Knowledge*.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. de las, Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. Le, Lavril, T., Wang, T., Lacroix, T., & Sayed, W. El. (2023). *Mistral 7B*. 1–9. http://arxiv.org/abs/2310.06825

Jonatan, V., & Igor, and A.-A. (2023). Creation of a Chatbot Based on Natural Language Processing for Whatsapp. *Journal of Database Management*, *3*(4), 39–53. https://doi.org/10.14810/elelij.2023.12402

Joshi, M., Choi, E., Weld, D. S., & Zettlemoyer, L. (2017). TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, *1*, 1601–1611. https://doi.org/10.18653/v1/P17-1147

Khan, A. A. (2023). *“Crafting Conversational Magic: Building an AI Language Model(LLM) App with Langchain?and Gradio.”* Medium.Com. https://medium.com/@abirkhan4u/crafting-conversational-magic-building-an-ai-language-model-llm-app-with-langchain-and-gradio-496429567512

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M. W., Dai, A. M., Uszkoreit, J., Le, Q., & Petrov, S. (2019). Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics*, *7*, 453–466. https://doi.org/10.1162/tacl\_a\_00276

Lin, C.-Y. (2004). *ROUGE: A Package for Automatic Evaluation of Summaries Chin-Yew*. 74–81. https://doi.org/10.1253/jcj.34.1213

Loukas, L., Stogiannidis, I., Malakasiotis, P., & Vassos, S. (2023). Breaking the Bank with ChatGPT: Few-Shot Text Classification for Finance. *FinNLP-Muffin 2023 - Joint Workshop of the 5th Financial Technology and Natural Language Processing and 2nd Multimodal AI For Financial Forecasting, in Conjunction with IJCAI 2023 - Proceedings*, 74–80.

Martin, M. (2023). *Large Language Models (LLMs): Definition, Examples, and Benefits*. Smith.Ai. https://smith.ai/blog/llm-ai

Mihaylov, T., Clark, P., Khot, T., & Sabharwal, A. (2018). Can a suit of armor conduct electricity? A new dataset for open book question answering. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, 2381–2391. https://doi.org/10.18653/v1/d18-1260

Minsky, M. (1989). *Steps Toward Artificial Intelligence*. Web.Media.Mit.Edu. https://web.media.mit.edu/~minsky/papers/steps.html

Mohamadi, S., Mujtaba, G., Le, N., Doretto, G., & Adjeroh, D. A. (2023). *ChatGPT in the Age of Generative AI and Large Language Models: A Concise Survey*. 1–60. http://arxiv.org/abs/2307.04251

Neves, M. C. (2023). *What are Quantized LLMs?* Https://Www.Tensorops.Ai/.

Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don’t know: Unanswerable questions for SQuAD. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, *2*, 784–789. https://doi.org/10.18653/v1/p18-2124

Rakotoson, Loïc, Massip, Sylvain, A. A. Laleye, F. (2024). *Retrieval Augmented Generation Architecture*. Www.Researchgate.Net.

Revou.co. (2024). *Apa itu Natural Language Processing (NLP)*. Revou.Co. https://revou.co/kosakata/natural-language-processing

Risyad, S. A. (2023). *Data Set: Pengertian, Jenis, dan Contohnya*. Dibimbing.Id. https://dibimbing.id/blog/detail/pengertian-data-sheet-jenis-dan-contoh

Russel, Stuart J; Norvig, P. (2010). *Artificial intelligence :a modern approach*.

Sakaguchi, K., Le Bras, R., Bhagavatula, C., & Choi, Y. (2020). WINOGRANDE: An adversarial winograd schema challenge at scale. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, 8732–8734. https://doi.org/10.1609/aaai.v34i05.6399

Sap, M., Rashkin, H., Chen, D., Le Bras, R., & Choi, Y. (2019). Social IQA: Commonsense reasoning about social interactions. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 4463–4473. https://doi.org/10.18653/v1/d19-1454

Suzgun, M., Scales, N., Schärli, N., Gehrmann, S., Tay, Y., Chung, H. W., Chowdhery, A., Le, Q. V., Chi, E. H., Zhou, D., & Wei, J. (2023). Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 13003–13051. https://doi.org/10.18653/v1/2023.findings-acl.824

Talmor, A., Herzig, J., Lourie, N., & Berant, J. (2019). CommonSenseqa: A question answering challenge targeting commonsense knowledge. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *1*, 4149–4158.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., … Scialom, T. (2023). *Llama 2: Open Foundation and Fine-Tuned Chat Models*. http://arxiv.org/abs/2307.09288

Trozze, A., Davies, T., & Kleinberg, B. (2023). *Large Language Models in Cryptocurrency Securities Cases: Can ChatGPT Replace Lawyers?* 1–49. http://arxiv.org/abs/2308.06032

ult.usk.ac.id. (2024). *Unit Layanan Terpadu USK*. Ult.Usk.Ac.Id.

V., K. (2024). *Fine-Tuning Large Language Models with Unsloth*. Medium.Com. https://medium.com/@kushalvala/fine-tuning-large-language-models-with-unsloth-380216a76108

Walker II, S. M. (2024). *What is the ROUGE Score (Recall-Oriented Understudy for Gisting Evaluation)?* Klu.Ai. https://klu.ai/glossary/rouge-score

Wicaksono, K. T. (2023). *Memahami Retrieval Augmented Generation (RAG) Melalui Rencana Pembangunan LaporMacet.com*. Medium.Com. https://kukuhtw.medium.com/memahami-retrieval-augmented-generation-rag-melalui-rencana-pembangunan-lapormacet-com-5990a76edda7

www.gradio.app. (2024). *Gradio*. Www.Gradio.App.

www.wiz.ai. (2023). *Menelaah : Bagaimana LLM (Large Language Models) bekerja*. Www.Wiz.Ai.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., & Choi, Y. (2020). Hellaswag: Can a machine really finish your sentence? *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 4791–4800. https://doi.org/10.18653/v1/p19-1472

Zhao, H., Ling, Q., Pan, Y., Zhong, T., Hu, J.-Y., Yao, J., Xiao, F., Xiao, Z., Zhang, Y., Xu, S.-H., Wu, S.-N., Kang, M., Wu, Z., Liu, Z., Jiang, X., Liu, T., & Shao, Y. (2023). Ophtha-LLaMA2: A Large Language Model for Ophthalmology. *ArXiv [Preprint]*, 1–19. http://arxiv.org/abs/2312.04906

Zhong, W., Cui, R., Guo, Y., Liang, Y., Lu, S., Wang, Y., Saied, A., Chen, W., & Duan, N. (2023). *AGIEval: A Human-Centric Benchmark for Evaluating Foundation Models*. http://arxiv.org/abs/2304.06364