BAB 1
PENDAHULUAN

1.1 Latar Belakang Masalah

Seiring dengan perkembangan teknologi informasi, kebutuhan pengguna terhadap akses informasi yang cepat, akurat, dan efisien semakin meningkat. Perkembangan *Artificial Intelligence* (AI), khususnya pada bidang *Natural Language Processing* (NLP) dan *Large Language Models* (LLM), telah menghadirkan sistem yang mampu memahami pertanyaan dalam bahasa alami serta memberikan respons secara interaktif yang menyerupai manusia [1, 2]. Kondisi ini mendorong perubahan fundamental terhadap ekspektasi pengguna dalam mengakses informasi [1, 2]. Pengguna tidak lagi hanya membutuhkan ketersediaan daftar tautan informasi, tetapi juga mengharapkan agen percakapan (*conversational agent*) yang mampu menyajikan jawaban secara langsung, relevan, dan sesuai konteks dalam waktu singkat [1].

Kebutuhan tersebut juga terlihat dalam lingkungan perguruan tinggi, khususnya dalam akses terhadap informasi akademik. Perguruan tinggi umumnya telah menyediakan berbagai informasi penting melalui situs web resmi, seperti panduan akademik, *handbook*, prosedur administrasi, pengumuman, serta dokumen pendukung lainnya. Namun, pada praktiknya, informasi tersebut sering kali disajikan dalam banyak file dokumen (*siloed data*) yang terpisah-pisah [3]. Kondisi ini menyebabkan mahasiswa harus mencari file-file dokumen tersebut secara manual untuk menemukan informasi yang dibutuhkan, sebuah proses yang tidak efisien dan rentan terhadap kelebihan informasi (*information overload*).

Di sisi lain, model bahasa besar komersial, seperti ChatGPT, memiliki kemampuan yang sangat baik dalam memahami pertanyaan dan menghasilkan jawaban secara natural [1, 4]. Akan tetapi, model semacam ini pada dasarnya tidak memiliki akses langsung terhadap pengetahuan internal, domain spesifik, atau dokumen berpemilik (*proprietary documents*) milik suatu institusi [5]. Akibatnya, ketika dihadapkan pada pertanyaan yang membutuhkan fakta spesifik dari kampus, model LLM rentan mengalami fenomena halusinasi (*hallucination*), yaitu kondisi saat model menghasilkan jawaban yang terdengar meyakinkan, tetapi secara faktual salah atau tidak bersumber [5, 6]. Oleh karena itu, diperlukan suatu pendekatan eksternal yang memungkinkan model bahasa mengakses dan memanfaatkan

sumber pengetahuan internal institusi tanpa harus melakukan pelatihan ulang model (*fine-tuning*) yang memakan biaya besar [3]. Pendekatan *fine-tuning* membutuhkan dataset pelatihan yang besar, biaya komputasi yang tinggi, serta proses pelatihan ulang setiap kali data diperbarui [3]. Sementara itu, mesin pencari konvensional hanya mengembalikan tautan dokumen tanpa memberikan jawaban langsung yang kontekstual. Di sisi lain, *Retrieval-Augmented Generation* (RAG) memungkinkan LLM menghasilkan jawaban yang tetap berbasis dokumen sumber tanpa memerlukan pelatihan ulang model, sehingga lebih efisien dan adaptif terhadap pembaruan data [3].

Salah satu pendekatan mutakhir yang banyak digunakan untuk menjawab kebutuhan tersebut adalah *Retrieval-Augmented Generation* (RAG). RAG adalah metode yang menggabungkan proses pencarian informasi spesifik dari basis data dokumen eksternal dengan kemampuan generatif model bahasa untuk menghasilkan jawaban yang lebih relevan, faktual, dan dapat ditelusuri sumbernya [7, 8]. Melalui pendekatan ini, sistem tidak murni bergantung pada memori internal model (pengetahuan parametrik), melainkan memanfaatkan potongan dokumen yang secara dinamis dimasukkan ke dalam konteks model (pengetahuan non-parametrik) [7]. Dalam konteks situs web kampus, pendekatan RAG memungkinkan berbagai dokumen akademik diintegrasikan ke dalam satu sistem *knowledge base* terpusat, sehingga bot dapat menjawab pertanyaan mahasiswa berdasarkan buku panduan resmi kampus.

Dalam merancang arsitektur RAG yang andal, kualitas jawaban akhir sangat dipengaruhi oleh komponen pencarian (*retrieval*), karena konteks yang salah atau tidak relevan dapat memicu kegagalan sistem (*failure points*) [9]. Salah satu tahapan paling krusial dalam *retrieval* adalah *chunking*, yaitu proses pemecahan dokumen panjang menjadi potongan-potongan teks yang lebih kecil agar dapat diproses secara efisien oleh model bahasa yang memiliki batasan "jendela konteks" (*context window*) [3]. Proses *chunking* dapat diimplementasikan menggunakan *Recursive Character Text Splitting* melalui pustaka LangChain, sehingga dokumen dipotong berdasarkan ukuran *chunk* dan *overlap* tertentu. Strategi penentuan ukuran *chunk* sangat menentukan kualitas semantik informasi [3]. Apabila ukuran *chunk* terlalu kecil, kalimat akan terpotong sehingga makna utuh informasi hilang. Sebaliknya, apabila ukuran *chunk* terlalu besar, potongan tersebut dapat menyertakan banyak kalimat yang tidak relevan (kebisingan/noise) yang akan memperlambat waktu komputasi inferensi model [3]. Pemilihan pendekatan *retrieval*-based seperti RAG alih-alih sekadar memperluas *context window* juga didukung oleh temuan Xu et

al. yang menunjukkan bahwa augmentasi *retrieval* dapat mencapai performa yang setara dengan model ber*context window* panjang namun dengan biaya komputasi yang jauh lebih rendah [10].

Selain strategi ukuran *chunk*, parameter *top-k retrieval* juga memiliki peran signifikan dalam menentukan keakuratan sistem RAG [11]. Parameter ini berfungsi membatasi jumlah potongan dokumen (*chunks*) teratas yang dianggap paling relevan terhadap pertanyaan pengguna untuk dimasukkan ke dalam konteks LLM. Pada sistem RAG modern, baik kueri pengguna maupun potongan dokumen umumnya terlebih dahulu diubah menjadi vektor agar pencarian semantik dapat dilakukan secara efisien pada basis data vektor. Secara intuitif, memberikan banyak dokumen (nilai *top-k* besar) tampaknya menguntungkan. Namun, penelitian terbaru menemukan paradoks bahwa mengekspos LLM pada terlalu banyak potongan dokumen justru berpotensi menurunkan akurasi jawaban akibat masuknya konteks yang tidak relevan [11, 12]. Sebaliknya, nilai *top-k* yang terlalu ketat (kecil) berisiko menyebabkan dokumen yang memuat informasi krusial tidak terjangkau dalam hasil pencarian (*missed the top ranked documents*) [9], sehingga informasi yang sebenarnya dibutuhkan untuk menjawab pertanyaan dengan benar menjadi hilang.

Berbagai penelitian terdahulu telah berupaya menyoroti analisis dan penyempurnaan arsitektur RAG secara terpisah. Sebagai contoh konkret, eksperimen oleh Gao et al. (2023) menunjukkan bahwa *chunk* yang tidak dioptimalkan dapat menurunkan retensi konteks [3]. Di sisi lain, Cuconasu et al. (2024) dan Yoran et al. (2024) meneliti dampak parameter *top-k* dan membuktikan bahwa pemenuhan *prompt* dengan terlalu banyak dokumen yang mengganggu (*distracting documents*) justru dapat menurunkan performa LLM [11, 12]. Meskipun literatur-literatur tersebut mengonfirmasi kerentanan parameter RAG, penelitian yang secara sistematis menguji **kombinasi kedua komponen tersebut secara silang dan bersamaan** pada dokumen institusional yang berstruktur hierarkis, seperti panduan akademik kampus, masih sangat terbatas. Padahal, pada lingkungan implementasi nyata di perguruan tinggi, efektivitas pencarian informasi sangat bergantung pada penemuan konfigurasi terbaik dari rasio ukuran *chunk* dan batas penarikan dokumen (*top-k*) agar sistem terhindar dari pelemahan konteks akibat masuknya informasi yang tidak relevan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah sebagai berikut.

1. Bagaimana sistem *Retrieval-Augmented Generation* (RAG) pada *chatbot* akademik berbasis web dapat diimplementasikan?
2. Kombinasi parameter *chunking* dan *top-k retrieval* mana yang menghasilkan performa *retrieval*, seperti *Hit Rate* dan *Mean Reciprocal Rank*, serta Akurasi Faktual jawaban terbaik pada sistem RAG yang diimplementasikan tersebut?

1.3 Batasan Permasalahan

Agar penelitian ini lebih terfokus dan terarah, batasan masalah dalam penelitian ini ditetapkan sebagai berikut.

1. Penelitian ini mencakup implementasi sistem *Retrieval-Augmented Generation* (RAG) berbasis web yang diberi nama ASKU, sekaligus pengujian eksperimental terhadap parameter *chunking* dan *top-k retrieval* pada sistem tersebut. Nama ASKU dibentuk dari kata *ask* dalam bahasa Inggris yang berarti bertanya dan huruf *U* yang merujuk pada Universitas Multimedia Nusantara (UMN), sehingga nama tersebut merepresentasikan fungsi sistem sebagai sarana tanya jawab akademik di lingkungan UMN. Penelitian ini tidak membahas perancangan dan pengembangan antarmuka pengguna (*User Interface*) secara mendalam.
2. Penelitian ini tidak melakukan pelatihan ulang (*fine-tuning*) maupun pengembangan model *Artificial Intelligence* (AI) baru secara langsung, melainkan memanfaatkan model bahasa besar yang sudah tersedia (*pre-trained*).
3. Sumber pengetahuan eksternal yang digunakan dalam sistem dibatasi pada dokumen akademik yang tersedia di MyUMN. Dokumen tersebut meliputi panduan akademik, prosedur administrasi, serta informasi lain yang berkaitan dengan kebutuhan mahasiswa.
4. Penelitian ini hanya memfokuskan pengujian pada strategi *chunking* dokumen dan parameter *top-k retrieval* sebagai variabel yang dianalisis dalam upaya meningkatkan kualitas jawaban pada sistem tanya jawab berbasis web.

5. Evaluasi performa sistem dilakukan secara kuantitatif berdasarkan hasil pengukuran *Hit Rate*, *Mean Reciprocal Rank* (MRR), dan Akurasi Faktual. Penelitian ini tidak mengevaluasi aspek teknis infrastruktur, seperti penggunaan RAM/CPU dan latensi jaringan, serta tidak mengukur tingkat kepuasan pengguna akhir melalui *User Acceptance Test* (UAT) atau kuesioner pasca-implementasi.

6. Model bahasa dasar (LLM) yang digunakan sebagai generator jawaban dan reformulasi pertanyaan dibatasi pada `gpt-4o-mini`. Konversi teks menjadi vektor dibatasi menggunakan model *embedding* Gemini yang diakses melalui penyedia agregator OpenRouter dengan identifier `google/gemini-embedding-2`; vektor yang dihasilkan diseragamkan menjadi 1024 dimensi sebelum disimpan dan dicari pada Supabase PostgreSQL dengan ekstensi `pgvector`. Penelitian ini tidak melakukan studi komparasi performa antar berbagai vendor LLM maupun model *embedding* di pasaran.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, tujuan penelitian ini adalah sebagai berikut.

1. Mengimplementasikan sistem *Retrieval-Augmented Generation* (RAG) pada *chatbot* akademik berbasis web bernama ASKU.

2. Menentukan kombinasi parameter *chunking* dan *top-k retrieval* yang menghasilkan performa terbaik pada sistem ASKU berdasarkan metrik *Hit Rate*, *Mean Reciprocal Rank* (MRR), dan Akurasi Faktual.

1.5 Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah sebagai berikut.

1. Bagi Pengembangan Keilmuan (Manfaat Teoritis): Penelitian ini diharapkan dapat memperkaya literatur dan menjadi referensi bagi penelitian selanjutnya terkait implementasi sistem *Retrieval-Augmented Generation* (RAG), khususnya mengenai pengaruh *chunking* dan *top-k retrieval* terhadap kualitas *retrieval* dan jawaban.

2. Bagi Institusi Perguruan Tinggi (Manfaat Praktis): Hasil penelitian ini memberikan rekomendasi teknis berbasis data mengenai konfigurasi parameter RAG terbaik pada sistem berbasis ASKU. Hal ini dapat berguna bagi institusi kampus yang ingin mengimplementasikan *chatbot* AI akademik tanpa harus melakukan pelatihan ulang model bahasa.

3. Bagi Mahasiswa (Manfaat Praktis): Penelitian ini mendemonstrasikan pemanfaatan ASKU sebagai prototipe sistem tanya jawab akademik berbasis RAG yang memungkinkan mahasiswa memperoleh jawaban akademik yang relevan, faktual, dan cepat berdasarkan dokumen resmi kampus.

1.6 Sistematika Penulisan

Sistematika penulisan laporan penelitian ini disusun untuk memberikan gambaran ringkas mengenai alur pembahasan dari Bab 1 hingga Bab 5. Adapun sistematika penulisan laporan adalah sebagai berikut.

1. Bab 1 PENDAHULUAN
   Bab ini memuat latar belakang masalah mengenai kebutuhan akses informasi akademik di lingkungan UMN serta urgensi pendekatan *Retrieval-Augmented Generation* (RAG) untuk mereduksi risiko halusinasi *Large Language Model*, rumusan masalah terkait implementasi sistem dan kombinasi parameter *chunking* serta *top-k retrieval* yang optimal, batasan permasalahan, tujuan dan manfaat penelitian, serta sistematika penulisan laporan.

2. Bab 2 LANDASAN TEORI
   Bab ini berisi landasan teori yang mendukung penelitian, meliputi konsep *Artificial Intelligence*, *Natural Language Processing*, *Large Language Models*, *Retrieval-Augmented Generation*, *chunking*, *embedding*, pencarian vektor, metrik evaluasi, serta penelitian terdahulu yang relevan.

3. Bab 3 METODOLOGI PENELITIAN
   Bab ini menjelaskan metodologi penelitian yang digunakan, mencakup jenis dan pendekatan penelitian, variabel penelitian, sumber dan prapemrosesan data, implementasi dan arsitektur sistem ASKU, desain eksperimen, serta prosedur evaluasi yang diterapkan.

4. Bab 4 HASIL DAN DISKUSI
   Bab ini menyajikan implementasi sistem ASKU, hasil eksperimen yang

diperoleh dari pengujian parameter *chunking* dan *top-k retrieval*, disertai analisis hasil, pembahasan temuan eksperimen, serta identifikasi konfigurasi terbaik sistem.

5. Bab 5 SIMPULAN DAN SARAN
   Bab ini berisi simpulan terkait keberhasilan implementasi sistem RAG ASKU serta konfigurasi parameter terbaik yang diperoleh (*chunk size* 1500, *overlap* 10%, dan *top-k* 5), serta saran untuk pengembangan penelitian selanjutnya seperti eksplorasi teknik *Advanced RAG*, pengujian dengan lebih dari satu dokumen, dan evaluasi variasi formulasi pertanyaan.