Penerapan Retrieval-Augmented Generation untuk Pembuatan Soal Pilihan Ganda dari Materi Berbasis PDF

K S Handoyo[*1], M L Klau[2], R J Wijaya[3] and R P Kristianto[4]

[1-4]Universitas Katolik Darma Cendika, Indonesia

E-mail: kevin.handoyo@student.ukdc.ac.id[1], mario.klau@student.ukdc.ac.d[2], ricky.wijaya@student.ukdc.ac.id[3], ryan.kristianto@ukdc.ac.id

Abstrak. Pembuatan soal pilihan ganda merupakan bagian penting dalam evaluasi pembelajaran, namun penyusunannya masih memerlukan waktu dan konsistensi yang tinggi. Pemanfaatan Large Language Model (LLM) membuka peluang otomatisasi, tetapi generasi soal tanpa konteks dokumen sering menghasilkan soal yang kurang relevan terhadap materi sumber. Penelitian ini menerapkan pendekatan Retrieval-Augmented Generation (RAG) untuk menghasilkan soal pilihan ganda secara otomatis dari dokumen pembelajaran berformat PDF. Metode yang digunakan meliputi ekstraksi teks dokumen, chunking teks, pembentukan embedding, penyimpanan vektor, mekanisme retrieval berbasis kemiripan semantik, serta generasi soal menggunakan model GPT-4o-mini. Dataset terdiri dari dokumen buku teks SMA/SMK kelas XI dari beberapa mata pelajaran. Evaluasi dilakukan menggunakan metrik ROUGE untuk mengukur kesesuaian konten antara soal hasil generasi dan soal referensi. Hasil evaluasi menunjukkan nilai ROUGE-1 sebesar 0,78, ROUGE-2 sebesar 0,63, dan ROUGE-L sebesar 0,78, yang mengindikasikan bahwa sistem mampu menghasilkan soal dengan tingkat relevansi dan kesesuaian konteks yang baik terhadap materi sumber. Temuan ini menunjukkan bahwa pendekatan RAG efektif dalam menjaga keterikatan konteks dan meningkatkan kualitas generasi soal otomatis berbasis dokumen pembelajaran.

Kata kunci: retrieval-augmented generation; generasi soal otomatis; large language model; rouge; pdf

Abstract. Multiple-choice question generation plays a crucial role in educational assessment, yet it often requires significant time and consistency. Large Language Models (LLMs) enable automation; however, question generation without document context frequently results in low relevance. This study applies a Retrieval-Augmented Generation (RAG) approach to automatically generate multiple-choice questions from PDF-based learning materials. The proposed method includes document text extraction, text chunking, embedding generation, vector storage, semantic similarity-based retrieval, and question generation using the GPT-4o-mini model. The dataset consists of senior high school textbooks from multiple subjects. Evaluation is conducted using ROUGE metrics to measure content similarity between generated questions and reference questions. The experimental results show ROUGE-1, ROUGE-2, and ROUGE-L scores of 0.78, 0.63, and 0.78, respectively, indicating that the system effectively maintains contextual relevance and semantic alignment with the source material. These findings demonstrate that the RAG approach improves the quality and contextual grounding of automatic question generation from educational documents.

Keywords: retrieval-augmented generation; automatic question generation; large language model, rouge; pdf

1. Pendahuluan

Proses evaluasi pembelajaran merupakan instrumen yang memiliki peran strategis dalam menilai efektivitas proses pendidikan, pendidik dapat memperoleh informasi objektif mengenai pencapaian hasil belajar peserta didik [1]. Selain mampu menggambarkan capaian belajar siswa, instrumen tersebut juga harus dirancang secara efektif, relevan dengan materi, dan sesuai dengan tujuan pembelajaran [2]. Salah satu metode yang sering digunakan dalam evaluasi adalah soal pilihan ganda, peserta didik akan diberikan sejumlah jawaban potensial dan diminta untuk memilih satu yang paling akurat menurut mereka yang mana akan melatih objektivitas dan pemahaman siswa [3].

Terdapat tiga aspek atau kaidah yang perlu diperhatikan dalam pembuatan soal, pertama yaitu aspek materi agar soal yang dibuat benar-benar sesuai dengan kompetensi dan tidak keluar dari batas materi, lalu kedua ada aspek konstruksi agar petunjuk pengerjaan harus jelas dan mudah dipahami siswa, dan ketiga ada aspek bahasa dimana soal harus disusun dengan kalimat yang jelas, komunikatif, dan tidak ambigu [4]. Menurut penelitian oleh [5], masih ditemukan masalah dimana guru mengalami kendala dalam memilih stimulus soal yang relevan dengan materi untuk memunculkan masalah yang disajikan pada soal. Berdasarkan fakta lain di lapangan, sebagian besar guru masih mengalami kendala dalam menyusun soal, dan hasil supervisi perangkat pembelajaran dari 8 guru ditemukan bahwa mayoritas soal yang dibuat masih jauh dari kaidah penyusunan soal yang baik seperti soal yang tidak sesuai dengan materi yang dipelajari siswa, permasalahan atau instruksi yang sulit dipahami, dan soal dengan bahasa yang membingungkan serta sulit dimengerti siswa [6].

Adanya permasalahan tersebut yang dialami oleh banyak guru, maka dibutuhkan adanya otomatisasi yang mendukung pembuatan soal terutama soal pilihan ganda dengan mengikuti kaidah pembuatan soal. Dengan perkembangan teknologi kecerdasan buatan terutama di bidang *Natural Language Processing* (NLP) dan *Generative Artificial Intelligence*, membuka peluang untuk mengotomatisasi pembuatan soal dari teks menjadi pertanyaan beserta jawaban dengan lebih cepat dan efektif [7]. Berbagai penelitian terdahulu menunjukkan bahwa model bahasa berbasis *Deep Learning* mampu menghasilkan soal secara otomatis dari dokumen materi pembelajaran. Seperti penelitian terdahulu oleh [8], yang mengembangkan *Automatic Question Generator* (AQG) dengan menggunakan metode NLP *rule-based* dan *Machine Learning* biasa untuk membuat soal Biologi SMA dari *e-book*, meskipun kualitas soal yang dihasilkan belum optimal dengan hanya 60-70% soal dinilai layak, bergantung pada *template* dan hanya mendukung satu *domain* saja. Lalu pada penelitian oleh [9], digunakan model *Large Language Model* (LLM) yaitu LLaMA 3.2 3B untuk menghasilkan soal pilihan ganda secara otomatis dari teks materi pembelajaran dengan *output* JSON yang berisi pertanyaan, opsi jawaban, dan kunci jawaban, hasil soal yang didapat memiliki tingkat relevansi jawaban yang cukup tinggi pada jumlah soal yanag sedikit dan memiliki keterbatasan berupa waktu respon yang meningkat serta skor relevansi yang menurun jika jumlah soal diperbanyak.

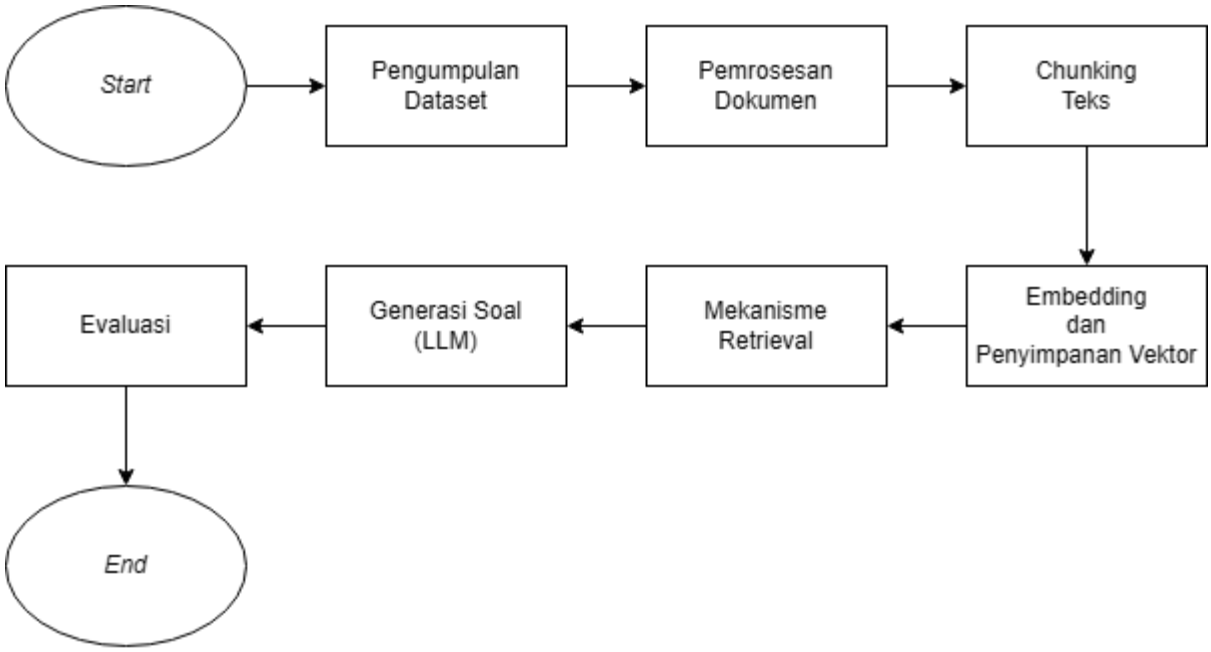
Berdasarkan keterbatasan pada beberapa penelitian terdahulu tersebut, NLP dan LLM murni biasa mampu menghasilkan soal secara otomatis, namun masih memerlukan pendekatan arsitektur yang lebih efisien dan terarah terhadap konteks dari materi. Salah satunya dengan mengintegrasikan mekanisme *Retrieval-Augmented Generation* (RAG) yang terlebih dahulu menyeleksi potongan-potongan informasi paling relevan dari kumpulan dokumen PDF sebelum digunakan [10]. Seperti pada penelitian terdahulu oleh [11], yang mengembangkan AQG berbasis RAG dengan memanfaatkan materi pembelajaran berformat PDF sebagai sumber pengetahuan, di mana teks materi dipecah menjadi potongan kecil dan diubah menjadi *embedding*, lalu disimpan dalam basis data vektor untuk mendukung proses pencarian konteks sebelum pembuatan soal oleh LLM (GPT-4o), penggunaan *retrieval* ini terbukti membantu meningkatkan ketajaman konteks dan mengurangi halusinasi model yang penting dalam pembuatan soal evaluasi.

Berdasarkan hal tersebut, dapat disimpulkan bahwa meskipun pendekatan NLP dan LLM telah menunjukkan potensi dalam otomatisasi pembuatan soal, masih terdapat tantangan dalam menjaga relevansi, konsistensi konteks, dan kesesuaian soal terhadap sumber materi pembelajaran. Oleh karena

itu, penelitian ini bertujuan untuk menerapkan pendekatan *Retreval-Augmented Generation* dalam menghasilkan soal pilihan ganda dari dokumen pembelajaran berformat PDF. Fokus penelitian ini adalah pada perancangan alur RAG yang mencakup proses pemrosesan dokumen, *chunking* teks, *embedding*, *retrieval* konteks, dan generasi soal serta analisis dari hasil soal tersebut. Diharapkan hasil penelitian ini dapat memberikan kontribusi dalam pengembangan sistem pembuatan soal otomatis yang lebih sesuai dengan kebutuhan evaluasi pembelajaran.

2. Metode Penelitian

Pada metode penelitian ini, akan dilakukan berbagai proses untuk mengembangkan RAG yang bisa menghasilkan soal pilihan ganda secara otomatis dari materi pembelajaran berbasis PDF. Proses yang dilakukan diawali dengan pengumpulan dataset berupa materi pembelajaran berformat PDF. Lalu di tahap kedua dilakukan pemrosesan dokumen PDF dengan mengekstraksi teks per halaman. Ditahap ketiga dilakukan proses *chunking* teks untuk memisah teks menjadi potongan-potongan kecil untuk menjaga kesinambungan teks. Setelah itu ditahap keempat dilakukan pembentukan *embedding* dan menyimpannya ke dalam basis data vektor. Lalu ditahap kelima baru diintegrasikan mekanisme *retrieval* dengan mengambil potongan teks paling relevan dengan topik atau konteks sehingga konteks soal yang diberikan tetap terikat pada sumber materi. Pada tahap terakhir dilakukan proses generasi soal dengan menggunakan LLM dengan *prompt* yang sudah dirancang agar menghasilkan sejumlah soal pilihan ganda yang sesuai. Pada Gambar 1 merupakan diagram alur kerja pada metode penelitian ini.



Gambar 1. Diagram Alur Kerja

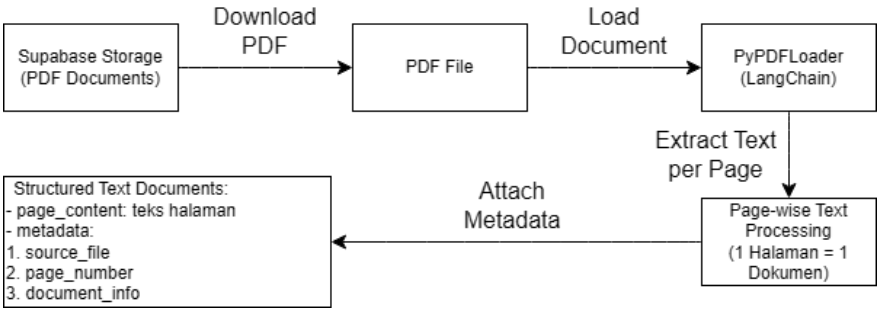
2.1. Dataset

Dataset yang digunakan berupa dokumen pembelajaran berformat PDF yang disimpan pada *database Supabase Object Storage*. Dokumen terdiri dari sembilan buku teks pembelajaran tingkat SMA/SMK kelas XI dari beberapa mata pelajaran, yaitu Biologi, Ekonomi, Fisika, Geografi, Informatika, Kimia, Matematika, Pendidikan Agama Buddha, dan Seni Musik. Dataset mencakup 9 dokumen PDF dengan total 2102 halaman, yang digunakan sebagai sumber pengetahuan atau *knowledge source* dalam sistem RAG.

2.2. Pemrosesan Dokumen

Tahap awal metode penelitian adalah pemrosesan dokumen PDF untuk mengekstraksi teks. Proses ini dilakukan menggunakan *tool* PyPDFLoader dari *library* LangChain. Setiap dokumen PDF diunduh dari *database Supabase Storage*, kemudian diekstraksi per halaman menjadi dokumen teks

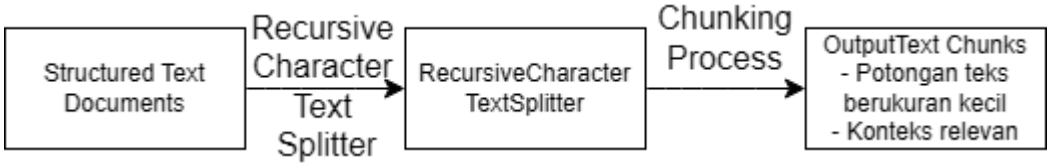
terstruktur. Metadata dokumen seperti sumber file, nomor halaman, dan informasi dokumen yang disertakan untuk menjaga keterlacakan konteks pada tahap *retrieval*. Tahap ini penting sebelum diterapkan ke dalam arsitektur RAG, karena kualitas teks dan pelabelan metadata akan berpengaruh langsung ke akurasi proses *retrieval* [12]. Pada Gambar 2 merupakan alur kerja dari pemrosesan dokumen.



Gambar 2. Alur Kerja Pemrosesan Dokumen

2.3. Chunking Teks

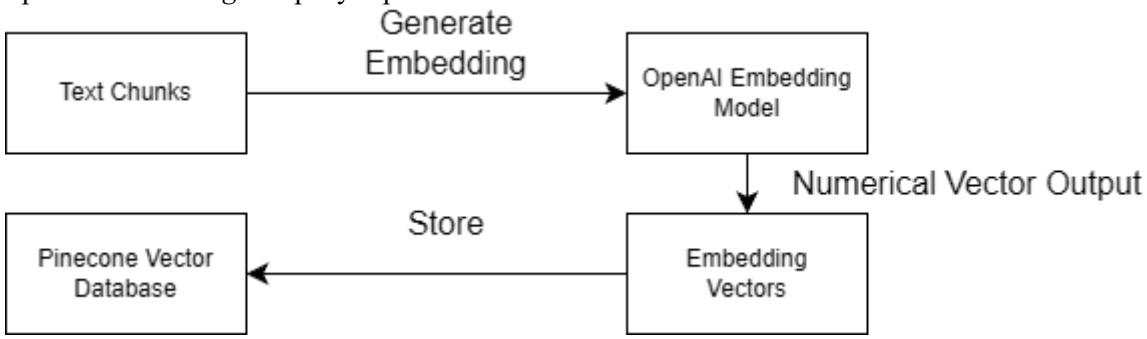
Teks hasil ekstraksi selajutnya dipecah menjadi potongan-potongan kecil atau *chunks* menggunakan *RecursiveCharacterTextSplitter*. *RecursiveCharacterTextSplitter* sendiri merupakan *node* untuk memecah data teks menjadi potongan yang bisa dikelola untuk meningkatkan efisiensi [13]. Pendekatan ini bertujuan untuk menjaga kesinambungan konteks antar potongan teks sekaligus menghindari kehilangan informasi penting pada batas *chunk*.



Gambar 3. Alur Kerja Chunking Teks

2.4. Embedding dan Penyimpanan Vektor

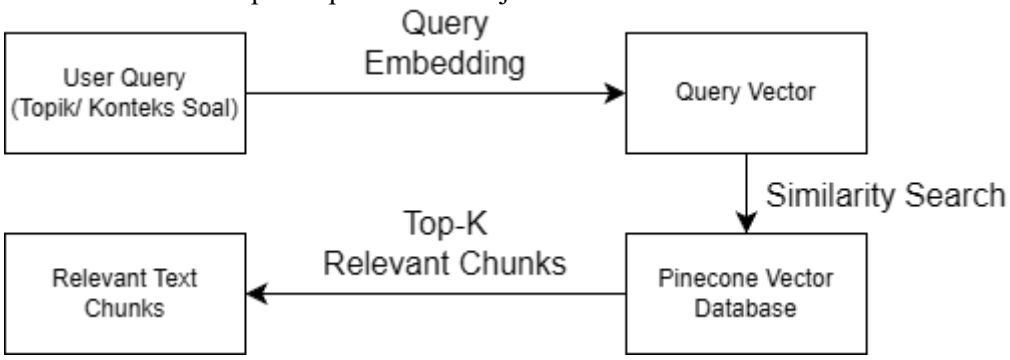
Setiap *chunk* teks diubah menjadi representasi vektor numerik menggunakan model *embedding* OpenAI *text-embedding-3-small* dengan dimensi vektor tertentu. Hasil *embedding* kemudian disimpan ke dalam *Pinecone Vector Database* menggunakan metrik *cosine similarity* untuk mendukung proses pencarian kemiripan semantik. Proses penyimpanan dilakukan secara bertahap untuk memastikan stabilitas dana efisiensi sistem saat memproses jumlah data yang besar. Gambar 4 merupakan alur kerja dari proses *Embedding* dan penyimpanan vektor.



Gambar 4. Alur Kerja Embedding dan Penyimpanan Vektor

2.5. Mekanisme Retrieval

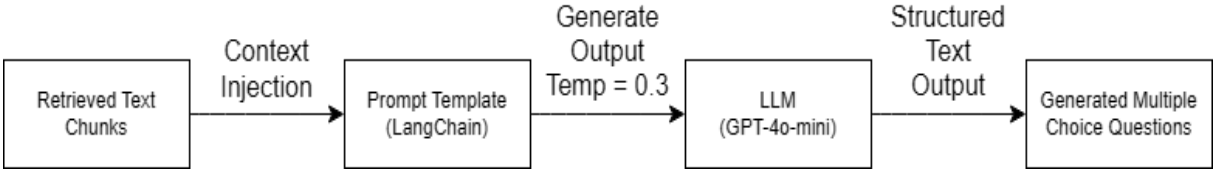
Pada tahap *retrieval*, sistem mengambil sejumlah potongan teks paling relevan dari *pinecone* bedasarkan *query* yang diberikan. *Query* ini merepresentasikan topik atau konteks soal yang akan dihasilkan. *Retrieval* sendiri merupakan teknologi yang menggabungkan kemampuan pencarian informasi dengan *Natural Language Processing* (NLP) [14]. Mekanisme *retrieval* berbasis *embedding* memungkinkan sistem untuk menemukan potongan materi yang memiliki kedekatan semantik tertinggi terhadap topik soal, sehingga konteks yang diberikan kepada model generatif tetap terikat pada materi sumber. Pada Gambar 5 merupakan proses alur kerja dari *Retrieval*.



Gambar 5. Alur Kerja Retrieval

2.6. Generasi Soal dengan LLM

Tahap akhir dalam alur RAG adalah generasi soal menggunakan *Large Language Model* GPT-4o-mini melalui *library* LangChain. Model menerima konteks hasil *retrieval* sebagai *input* tambahan sebelum melakukan generasi soal. *Prompt* dirancang untuk menghasilkan sejumlah soal pilihan ganda dengan empat opsi jawaban, dan satu kunci jawaban. Parameter *temperature* diatur pada nilai tertentu untuk menjaga keseimbangan antara variasi bahasa dan konsistensi konteks. *Output* dari model berupa teks soal yang terstruktur, mencakup pertanyaan, pilihan jawaban (A-D), dan kunci jawaban. Pada Gambar 6 merupakan alur kerja dari proses pembuatan soal.



Gambar 6. Alur Kerja Generasi Soal

2.7. Evaluasi

Pada tahap evaluasi ini, digunakan matrik *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE) untuk mengukur kualitas dari hasil soal yang sudah dibuat oleh RAG. ROUGE sendiri memang dirancang untuk mengevaluasi isi teks secara otomatis. ROUGE mendapatkan hasil skor dengan cara membandingkan urutan kata dengan teks hasil tulisan manusia. ROUGE terdiri dari tiga N yaitu ROUGE 1 untuk menghitung nilai satuk kata, ROUGE 2 untuk menghitung nilai dari dua kata yang berhubungan, dan ROUGE L yang membandingkan kata berurutan yang panjang pada dokumen [15]. Berikut ini merupakan rumus untuk menghitung ROUGE:

- *Precision* [15]:

$$ROUGE\ 1 = \frac{Jumlah\ unigram\ kata\ sama}{keseluruhan\ kata\ ringkasan\ sistem}$$

$$ROUGE\ 2 = \frac{Jumlah\ bigram\ kata\ sama}{keseluruhan\ kata\ ringkasan\ sistem}$$

$$ROUGE\ L = \frac{Longest\ Common\ Subsequence}{keseluruhan\ kata\ ringkasan\ sistem}$$

- *Recall* [15]:

$$ROUGE\ 1 = \frac{Jumlah\ unigram\ kata\ sama}{keseluruhan\ kata\ ringkasan\ manusia}$$

$$ROUGE\ 2 = \frac{Jumlah\ bigram\ kata\ sama}{keseluruhan\ kata\ ringkasan\ manusia}$$

$$ROUGE\ L = \frac{Longest\ Common\ Subsequence}{keseluruhan\ kata\ ringkasan\ manusia}$$

- *F1-score* [15]:

$$F1 - score = 2 \times \frac{precision + recall}{precision + recall}$$

3. Hasil dan Pembahasan

3.1. Hasil Pemrosesan Dokumen

Hasil dari tahap ini menunjukkan bahwa seluruh dokumen berhasil diekstraksi tanpa kehilangan struktur teks utama, sehingga konten materi pembelajaran dapat diproses lebih lanjut secara sistematis. Keberadaan metadata halaman juga memungkinkan sistem untuk melacak kembali sumber materi yang digunakan pada proses generasi soal, yang menjadi aspek penting dalam menjaga akurasi dan transparansi sistem RAG.

3.2. Hasil Chunking Teks

Teks hasil ekstraksi selajutnya dipecah menjadi potongan-potongan kecil atau *chunks* menggunakan *RecursiveCharacterTextSplitter*. Parameter *chunking* yang digunakan adalah:

- *Chunk size* : 250 karakter
- *Chunk overlap* : 75 karakter

Dari total 2102 halaman dokumen, dihasilkan sebanyak 19.460 *chunks* teks, yang selanjutnya dapat digunakan dalam proses *embedding*. Hasil *chunking* menunjukkan bahwa ukuran dan *overlap* yang digunakan mampu menghasilkan potongan teks yang relatif homogen serta masih mempertahankan konteks materi pembelajaran. Dengan demikian, setiap *chunk* memiliki representasi informasi yang cukup untuk mendukung proses pencarian semantik pada tahap *retrieval*.

3.3. Hasil Embedding dan Penyimpanan Vektor

Pada tahap ini setiap *chunk* teks diubah menjadi representasi vektor numerik menggunakan model *embedding* OpenAI *text-embedding-3-small* dengan dimensi vektor sebesar 1536. Hasil *embedding*

kemudian disimpan ke dalam *Pinecone Vector Database* menggunakan metrik *cosine similarity* untuk mendukung proses pencarian kemiripan semantik. Proses penyimpanan dilakukan secara bertahap untuk memastikan stabilitas dan efisiensi sistem saat memproses jumlah data yang besar. Dengan penyimpanan berbasis vektor ini, sistem mampu melakukan pencarian konteks materi secara cepat dan relevan berdasarkan *query* yang diberikan.

3.4. Hasil Mekanisme Retrieval

Pada tahap *retrieval*, sistem mengambil sejumlah potongan teks paling relevan dari *pinecone* bedasarkan *query* yang diberikan. Hasil *retrieval* menunjukkan bahwa potongan teks yang diambil memiliki keterkaitan langsung dengan materi sumber, sehingga konteks yang diberikan kepada model generatif tetap terikat pada dokumen pembelajaran. Hal ini membuktikan bahwa mekanisme *retrieval* dalam sistem RAG berperan penting dalam mengurangi kemungkinan munculnya informasi yang tidak relevan atau tidak bersumber.

3.5. Hasil Generasi Soal dengan LLM

Tahap generasi soal dilakukan menggunakan model LLM GPT-4o-mini dengan memanfaatkan konteks hasil *retrieval* dari sistem RAG. Model menerima potongan materi yang relevan sebagai konteks dan menghasilkan soal pilihan ganda sesuai dengan topik yang diminta. Berdasarkan hasil tersebut, model mampu menghasilkan soal dengan struktur yang konsisten, terdiri dari pertanyaan, empat opsi jawaban, dan satu kunci jawaban yang jelas. Selain itu, konten soal yang dihasilkan masih berada dalam lingkup materi dasar matematika yang sesuai dengan konteks yang diberikan oleh sistem RAG. Pada Tabel 1 merupakan contoh hasil generasi soal Matematika menggunakan model LLM GPT-4o-mini.

Tabel 1. Hasil Generasi Soal Matematika oleh Model LLM GPT-4o-mini

No	Soal	Opsi Jawaban	Kunci Jawaban
1	Jika \(x + 3 = 10 \), berapa nilai \(x \)?	A. 5 B. 7 C. 10 D. 13	B. 7
2	Sebuah segitiga memiliki panjang sisi 3 cm, 4 cm, dan 5 cm. Apa jenis segitiga tersebut?	A. Segitiga sama sisi B. Segitiga sama kaki C. Segitiga siku-siku D. Segitiga sembarang	C. Segitiga siku-siku
3	Berapa hasil dari \(15 \div 3 + 4 \times 2 \)?	A. 10 B. 12 C. 14 D. 16	B. 12

3.6. Hasil Evaluasi

Tahap ini evaluasi dilakukan untuk mengukur tingkat kesesuaian antara soal yang dihasilkan sistem dengan soal referensi menggunakan metrik ROUGE. Pada Tabel 2 dibawah ini merupakan hasil dari matrik evaluasi ROUGE.

Tabel 2. Hasil Evaluasi ROUGE

ROUGE	Precision	Recall	F1-score
ROUGE 1	0.7778	0.5000	0.6087
ROUGE 2	0.6250	0.3846	0.4762
ROUGE L	0.7778	0.5000	0.6087

Berdasarkan hasil evaluasi, nilai ROUGE-1 dan ROUGE-L masing-masing memperoleh nilai F1-score sebesar 0.6087, sedangkan ROUGE-2 memperoleh nilai F1-score sebesar 0.4762. Nilai tersebut menunjukkan bahwa soal yang dihasilkan memiliki tingkat kesamaan dan struktur kalimat yang cukup baik terhadap soal referensi meskipun tidak bersifat identik secara teks. Hasil ini mengindikasikan bahwa sistem mampu menghasilkan soal yang relevan secara konten, namun tetap memiliki variasi dalam penyusunan kalimat, yang merupakan karakteristik alami dari model generatif.

3.7. Analisis

Berdasarkan hasil pengujian yang telah dilakukan, penerapan dengan pendekatan *Retrieval-Augmented Generation* (RAG) menunjukkan kemampuan yang baik dalam menjaga relevansi dan keterikatan soal terhadap materi sumber. Mekanisme *retrieval* memungkinkan model generatif memperoleh konteks yang spesifik dan sesuai dengan topik, sehingga proses generasi soal menjadi lebih terarah dibandingkan dengan pendekatan generasi berbasis LLM tanpa dukungan dokumen. Hasil evaluasi menggunakan metrik ROUGE memperlihatkan bahwa soal yang dihasilkan memiliki tingkat kesamaan kata yang cukup dengan soal referensi, khususnya pada metrik ROUGE-1 dan ROUGE-L. Nilai tersebut mengindikasikan bahwa model mampu mempertahankan kata kunci dan struktur utama materi, sekaligus menghasilkan variasi kalimat melalui proses parafrase. Hal ini menunjukkan bahwa sistem tidak sekadar menyalin konten dari dokumen sumber, tetapi melakukan perubahan kata dengan tetap menjaga makna utama.

4. Kesimpulan dan Saran

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa pendekatan *Retrieval-Augmented Generation* (RAG) mampu diterapkan secara efektif untuk menghasilkan soal pilihan ganda secara otomatis dari dokumen pembelajaran berformat PDF. Integrasi mekanisme *retrieval* berbasis *embedding* dengan model *Large Language Model* (LLM) memungkinkan sistem untuk menghasilkan soal yang tetap terikat pada konteks materi sumber, sehingga relevansi dan konsistensi soal dapat terjaga. Hasil pengujian menunjukkan bahwa sistem RAG berhasil memproses dokumen pembelajaran dalam jumlah besar melalui tahapan pemrosesan dokumen, *chunking* teks, *embedding*, dan *retrieval* konteks sebelum proses generasi soal. Soal yang dihasilkan memiliki struktur yang konsisten, mencakup pertanyaan, opsi jawaban, dan kunci jawaban yang jelas, serta sesuai dengan lingkup materi yang diberikan. Evaluasi menggunakan metrik ROUGE menunjukkan bahwa soal yang dihasilkan memiliki tingkat kesamaan kata dan struktur kalimat yang cukup baik terhadap soal referensi, tanpa bersifat identik secara tekstual. Hal ini mengindikasikan bahwa sistem tidak hanya menyalin konten dari dokumen sumber, tetapi mampu melakukan parafrase dengan tetap mempertahankan makna utama materi. Dengan demikian, penerapan RAG dapat menjadi solusi yang potensial dalam mendukung otomatisasi pembuatan soal evaluasi pembelajaran yang lebih terarah dan berbasis konteks.

Penelitian selanjutnya disarankan untuk melakukan evaluasi kualitas soal secara pedagogis dengan melibatkan pakar pendidikan guna menilai kesesuaian tingkat kesulitan, kejelasan distraktor, dan ketercapaian tujuan pembelajaran. Selain itu, penggunaan metrik evaluasi tambahan di luar ROUGE

serta pengujian pada lebih banyak mata pelajaran dan jenjang pendidikan diharapkan dapat meningkatkan generalisasi dan kualitas sistem pembuatan soal berbasis RAG.

Referensi

[1] Handoko, F. Rohman, S. A. Farha, H. S. Putri, and D. A. Sucitra, "Optimalisasi Kualitas Butir Soal melalui Pelatihan Penyusunan Indikator dengan Uji Validitas dan Reliabilitas," *Jurnal Pengabdian Masyarakat Ilmu Pendidikan*, vol. 4, no. 2, pp. 296–305, Sep. 2025, doi: 10.23960/jpm-ip.vol.4i.2.1083.

[2] Suparji, Y. Anistyasari, and A. Wardhono, "Pelatihan Penyusunan Butir Soal Untuk Guru-guru Sekolah Indonesia Di Kuala Lumpur," Jan. 2025. doi: https://doi.org/10.36728/jpf.v6i1.4368.

[3] Muhammad Syauqi Firdaus, Wahyu Kholis Prihantoro, Latifaturrohmah Latifaturrohmah, Husna Rifaatul Mahmudah, and Afrida Aunil Illah, "Efektivitas Instrumen Tes Uraian Dibandingkan dengan Tes Pilihan Ganda dalam Mengukur Hasil Belajar Siswa di MAN 2 Bantul," *Jurnal Kajian Ilmu Pendidikan, Bahasa dan Komunikasi*, vol. 1, no. 4, pp. 87–101, Nov. 2025, doi: 10.61132/jkaipbaku.v1i4.175.

[4] S. Setiawati and M. Lapasau, "Aspek Bahasa dan Konstruksi Butir Soal Evaluasi Pada Buku Tematik Kelas III Sekolah Dasar," Jul. 2022. doi: https://doi.org/10.30998/sinastra.v1i0.6111.

[5] U. A. Sinta, G. Roebyanto, and N. L. S. Nuraini, "Analisis Kesulitan Guru dalam Menyusun Soal Evaluasi Berbasis Hots Pada Pembelajaran Matematika di SDN Torongrejo 2," *Jurnal Pembelajaran, Bimbingan, dan Pengelolaan Pendidikan*, vol. 2, no. 1, pp. 45–53, Jan. 2022, doi: 10.17977/um065v2i12022p45-53.

[6] D. Kasiono, S. Dasar, N. Sepanyul, K. Gudo, and K. Jombang, "Peningkatan Kemampuan Menyusun Soal Dengan Metode Pendampingan Berpola SP3R Pada Guru SDN Sepanyul Kecamatan Gudo Kabupaten Jombang Tahun 2018," 2019. doi: https://doi.org/10.26740/jdmp.v4n1.p33-41.

[7] M. F. Zain, "Penerapan Artificial Intelligence (AI) Dalam Pembuatan Soal Kuis di Aplikasi Andaliman Berbasis Learning Management System (LMS) Moodle," *WAWASAN: Jurnal Kediklatan Balai Diklat Keagamaan Jakarta*, vol. 5, no. 2, pp. 160–173, 2024, doi: https://doi.org/10.53800/8hc6dx24.

[8] J. Mardika, O. N. Pratiwi, and F. Hamami, "Automatic Question Generator Menggunakan Metode Template-Based," Apr. 2023.

[9] T. Dharmawan and A. Witanti, "Evaluasi LLAMA3.2 3B Untuk Menghasilkan Soal Otomatis dengan Deepeval Berdasarkan Metrik Answer Relevancy dan Hallucination," *Jurnal Informatika Teknologi dan Sains*, vol. 7, no. 1, pp. 242–248, Jan. 2025, doi: https://doi.org/10.51401/jinteks.v7i1.5423.

[10] I. Fanani, "Implementasi Retrieval Augmented Generation Untuk Evaluasi Proposal Tugas Akhir Mahasiswa," Mar. 2025. doi: https://doi.org/10.59820/tekomin.v3i2.336.

[11] Novri Rahman, N. S. Harahap, M. Affandes, and Pizaini, "Implementasi Langchain dan Large Language Models Dalam Automatic Question Generation Untuk Computer Assisted Test," *Bulletin of Computer Science Research*, vol. 5, no. 4, pp. 434–446, Jun. 2025, doi: 10.47065/bulletincsr.v5i4.558.

[12] W. Setyaningsih, "Implementasi Retrieval Augmented Generation Pada Information Retrieval And Response System (Studi Kasus: Anarkisme Emma Goldman) Menggunakan Langchain," Semarang, Feb. 2025. doi: https://repository.unissula.ac.id/40931/2/Teknik%20Informatika_32602000113_fullpdf.pdf?utm_source=chatgpt.com.

[13] Hoiriyah, M. Saidi Rahman, F. Ekawati, and Y. Indra Wijaya, "Chatbot AI Sebagai Media Pencarian Informasi dengan Menggunakan Metode Large Language Models (LLM)," May 2025. doi: https://doi.org/10.33084/jsakti.v7i2.9952.

[14] M. R. Rachman, M. Rosidin, and W. Y. Sulistyo, "Implementasi Metode Retrieval Augmented Generation Pada Chatbot Untuk Otomatisasi Layanan Pelanggan Kontrakan," *Jurnal teknik Informatika*, vol. 11, no. 02, p. 229, 2025.

[15] M. A. Rizky, “Analisis Efektivitas Dua Jenis Gaya Prompt dalam Model LLM Berbasis RAG,” *Jurnal Komtika (Komputasi dan Informatika)*, vol. 9, no. 1, pp. 76–86, May 2025, doi: 10.31603/komtika.v9i1.13488.