Automating Semantic Data Retrieval in Material Science Research Using RAG Frameworks

Nidhishree C, Panchami D
RV University, Bangalore, India
1RVU23CSE307, 1RVU23CSE325

***Abstract*—The rapid growth of material science research has led to an unprecedented surge in the volume and complexity of scientific data, encompassing experimental results, computational simulations, theoretical models, and industrial applications. Traditional keyword-based search methods, while widely used, are increasingly insufficient for extracting meaningful insights from such heterogeneous and semantically rich datasets. Researchers often face challenges in locating relevant information hidden within vast repositories of publications, patents, and databases, slowing down the discovery of novel materials and the advancement of sustainable technologies. Recent developments in Artificial Intelligence (AI) and Natural Language Processing (NLP) offer promising avenues to overcome these challenges. In particular, Retrieval-Augmented Generation (RAG) frameworks combine the strengths of Large Language Models (LLMs) with powerful information retrieval mechanisms to deliver context-aware, semantically accurate responses. Unlike conventional retrieval systems that rely on surface-level keyword matches, RAG-based approaches leverage embeddings and knowledge grounding to capture deeper semantic relationships across scientific literature and databases. This allows researchers to pose natural language queries and obtain precise, contextually enriched results that align with the intent of their inquiry. In this paper RAG framework has been explored for automating the process of semantic data retrieval particularly in the material science domain.**

***Index Terms*—RAG, Natural Language Processing, Semantic Search, Material Science, MatSciBERT, FAISS, Data Extraction**

I. Introduction

Materials science is an interdisciplinary field dedicated to studying the structure, composition, properties, and applications of materials for functional and behavioral needs. It integrates concepts from chemistry, physics, and engineering to understand the intrinsic behavior of matter and to drive the development of new materials [1]. The field encompasses a wide range of materials, including metals, ceramics, polymers, composites, and semiconductors, which are central to designing and advancing modern technologies. With increasing industrial demands, there is a growing need for stable, high-performance, and advanced materials in sectors such as electronics, battery technologies, aerospace, and healthcare. The discovery and optimization of such materials often depend on analyzing large volumes of experimental data, computational simulations, and scientific literature.

However, much of this information exists in unstructured formats such as PDFs, which are difficult to analyze and manage automatically because they lack a consistent schema or organization. Unstructured documents typically contain lengthy paragraphs, irregular formatting, dense technical language, and varied use of headings or tables, making it challenging to search, index, and extract relevant content. In contrast, structured data—such as databases, spreadsheets, or tabular datasets—follows a consistent format with clearly defined fields, making it easier to search, store, and analyze. To bridge this gap, sophisticated methods from Natural Language Processing (NLP) and Machine Learning (ML) are increasingly used to transform unstructured information into structured datasets. This conversion forms the basis for creating intelligent systems capable of efficient storage, search, and analysis of scientific knowledge.

Traditional knowledge retrieval approaches using large language models (LLMs) often fall short in this context. These models rely primarily on their pre-trained knowledge bases and cannot reliably process highly specialized, unstructured documents such as materials science PDFs. As a result, they may produce unverifiable or hallucinated responses.

To overcome these challenges, we propose a RAG-inspired pipeline that integrates domain-specific embeddings, efficient retrieval, and context grounding. At its core, the system leverages MatSciBERT, a pre-trained model specialized for materials science, to encode both user queries and document chunks into semantic vector embeddings. These embeddings capture the scientific meaning of words and phrases, ensuring accurate interpretation of technical terms (e.g., "synthesis" and "method" are recognized as semantically related). Extracted text from PDFs is processed into smaller, manageable chunks and indexed using FAISS, which supports scalable and efficient similarity search through approximate nearest neighbor algorithms. During retrieval, the query embedding is compared with indexed document embeddings using distance metrics such as cosine similarity or Euclidean distance, and the most relevant chunks are returned. This process ensures precise, context-aware matching even in large, complex datasets. Once the relevant excerpts are identified, they are combined with the user query and passed to Gemini, a large language model, which generates a coherent and context-grounded response. This approach ensures that the output is not only fluent but also supported by evidence from domain-specific documents. Additionally, all queries, retrieved excerpts, and outputs are logged for transparency and future analysis, making the system both auditable and reliable. [2].

In the following sections, we explore the key contributions

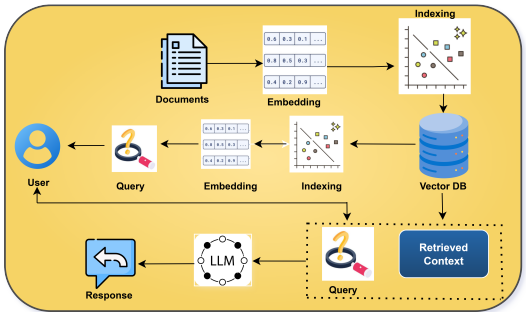

Fig. 1: RAG framework

this paper brings to light:

• Unlike traditional LLM-only retrieval, which is limited by static pretraining and lacks grounding in external knowledge, the proposed pipeline automates the process of searching, structuring, and reasoning over unstructured materials science data.

• By combining semantic embeddings, vector indexing, and retrieval-augmented generation (RAG), the system provides accurate, scalable, and domain-tailored knowledge retrieval.

• This makes it particularly effective for high-volume research tasks where scientists must navigate detailed information such as structural data, processing parameters, and performance characteristics. It also supports for downstream tasks by storing and organizing extracted information for long-term accessibility and trend analysis.

• Ultimately, the framework is capable of handling a wide variety of queries, from factual properties to conceptual insights which significantly reduces manual effort, accelerates knowledge discovery, and supports advanced research in materials science.

This paper is organized as follows: Section 2 gives a comprehensive literature survey of NLP in the field of material science. Section 3 describes the research methodology proposed for retrieving contextually relevant information. The findings are presented in Section 4, which emphasizes the efficiency of our system. Lastly, Section 5 concludes with a summary of findings and recommendations for future work in RAG applications.

II. Literature Review

Researchers across the world have increasingly adopted Retrieval-Augmented Generation (RAG) techniques for information retrieval from research papers, with the aim of accelerating scientific discovery. One such approach involves a specialized large language model (LLM)-based text generator designed specifically to handle queries related to metal additive manufacturing (AM), known as AMGPT

In [3], the authors explored novel technique to overcome the challenges that are encountered in Traditional Retrieval-Augmented Generation (RAG) approaches in Large Language Models (LLMs) such as outdated information, hallucinations, limited interpretability due to context constraints, and inaccurate retrieval. To address these issues, Graph RAG integrates graph databases to enhance the retrieval process. The authors proposed method processes Material Science documents by extracting key entities (referred to as MatIDs) from sentences, which are then utilized to query external Wikipedia knowledge bases (KBs) for additional relevant information. We implement an agent-based parsing technique to achieve a more detailed representation of the documents. Our improved version of Graph RAG called G-RAG further leverages a graph database to capture relationships between these entities, improving both retrieval accuracy and contextual understanding. This enhanced approach demonstrates significant improvements in performance for domains that require precise information retrieval, such as Material Science.

The authors of [4] performed a comparative study between several Natural Language Processing (NLP) strategies, such as the frequency-based method(spaCy), the transformer model (Simple T5), and retrieval augmented generation (RAG) with Large Language Model (GPT3.5-turbo). The SciTLDR dataset is chosen for this research experiment and the authors used three distinct techniques to implement three different systems for auto-generating the literature reviews. The ROUGE scores are used for the evaluation of all three systems. Based on the evaluation, the Large Language Model GPT-3.5-turbo achieved the highest ROUGE-1 score, 0.364. The transformer model comes in second place and spaCy is at the last position. Finally, a graphical user interface is created for the best system based on the large language model

III. Methodology

The workflow consists of several steps that assist in extracting, processing, retrieving and generating content-oriented details from PDFs in response to user queries. Query processing is the initial step in the pipeline, where user queries are analyzed and transformed into a format that allows for effective comparison with the stored document information. The pipeline handles both generic queries—such as those related to properties, synthesis methods, or applications—and custom queries defined by the user. A custom query refers to any user-defined input that seeks specific information not limited to pre-defined categories. Upon receiving a query, the pipeline first semantically analyzes and encodes it into a high-dimensional vector representation using the MatSciBert model. MatSciBERT is a pre-trained model adapted for materials science, designed to convert the query into a set of numerical values (embeddings). These embeddings capture the meaning of the words and phrases in the query within the scientific context.

The general architecture of a RAG system is shown in Figure 1, where documents are embedded, indexed in a vector database, and queried to retrieve relevant context before generating the final response.

A. Retrieval

The retrieval phase filters and isolates the most relevant information from a large collection of unstructured textual data, such as scientific PDFs. These documents are often

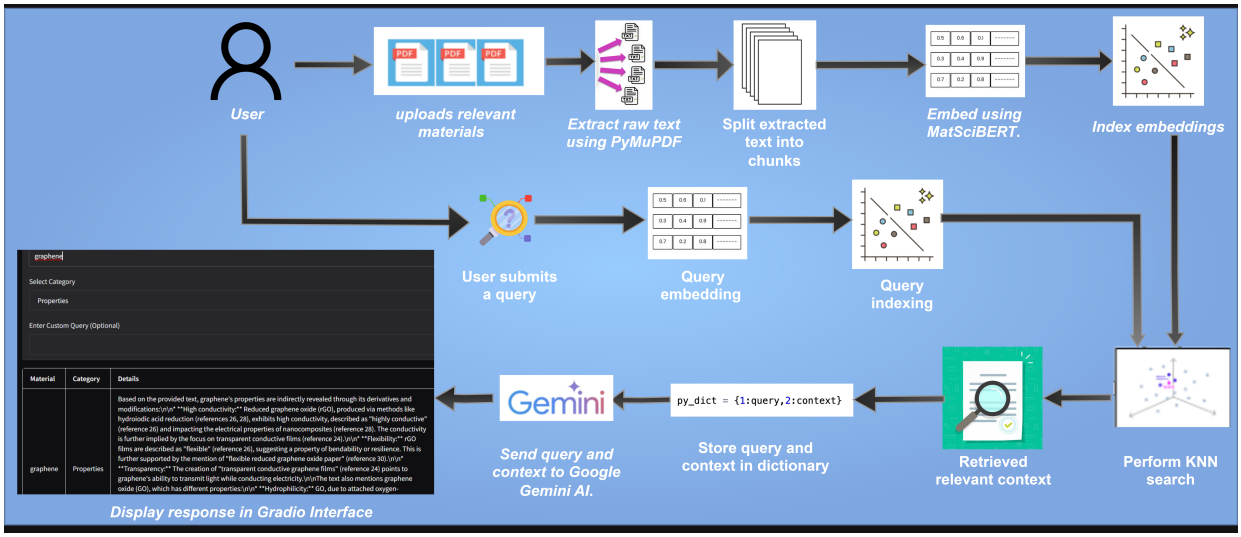

Fig. 2: End-to-End RAG Framework for Material Science Data Extraction and Query Response

not organized in a format suitable for direct querying. They typically contain lengthy paragraphs, varied formatting, inconsistent use of headings or tables, and dense technical language, which makes it difficult to pinpoint specific, context-relevant pieces of information directly. To address this, a series of steps are taken to semantically structure the content.

1) Data Ingestion: The pipeline begins by allowing users to upload multiple PDF documents. Each document is processed page by page, and the textual content of each page is systematically extracted with the help of PyMuPDF library. This ensures that the information from every section of the document is retained accurately. The extracted text serves as the foundational input for downstream processing, including chunking and embedding.

This approach ensures that even large and complex documents are handled efficiently, enabling accurate and scalable information retrieval from high-volume data sources.

2) Text Splitting: The proposed system splits the extracted text from each page into smaller, manageable chunks based on sentence boundaries. The text is first divided into individual sentences, and these sentences are then grouped together incrementally until a predefined character limit (e.g., 500 characters) is reached. If adding another sentence would exceed the limit, the current group is finalized as a chunk, and a new chunk is started. This method ensures that each chunk remains within size constraints while preserving sentence integrity, making the data suitable for downstream embedding and retrieval tasks.

Splitting the text this way also improves search efficiency and relevance by allowing the system to match smaller, more focused segments of text rather than large, ambiguous blocks.

3) Text Embedding: The role of embeddings in this system is critical to bridge the gap between the raw, unstructured text and the semantic query input. The text (such as a query or document) is processed through the embedding model MatSciBERT to generates embeddings by encoding the input into a fixed-length vector that captures the semantic relationships between the words and their context.

4) Indexing using FAISS: the system utilizes FAISS (Facebook AI Similarity Search), which is used for fast and scalable similarity search by indexing high-dimensional vectors and enabling approximate nearest neighbor retrieval. Rather than performing costly brute-force comparisons between the query embedding and all document embeddings, FAISS constructs an optimized index that enables fast similarity search via efficient distance computations. This allows the system to rapidly identify and return the most relevant text chunks, ensuring scalability and high performance even with millions of embeddings.

5) KNN Search: K-Nearest Neighbors (KNN) Search is a widely used technique for identifying the most similar data points within a dataset based on their positions in a vector space. The 'K' refers to the number of nearest neighbors to retrieve. By measuring the distance between vectors—using metrics like Euclidean distance or cosine similarity—the algorithm finds the K vectors that are closest to a given input vector, enabling efficient similarity-based retrieval across various applications.

In our pipleline, once the user's query is converted into an embedding, KNN is applied to search for the closest matches by calculating the distance (e.g., Euclidean distance) between the query embedding and the precomputed document embeddings in the vector space. However, instead of comparing the query embedding to every document embedding individually, we make use of the index to speed up the process. The index is essentially a pre-organized structure that groups similar embeddings, so when the query is passed through the KNN algorithm, it quickly narrows down the search space by focusing on the nearest embeddings in the index.

B. Generation

The pipeline begins by processing the user's query paragraph, which is parsed and segmented into meaningful components. To retrieve semantically relevant content, KNN algorithm is used over a FAISS-based vector index built from embedded text chunks of the input PDFs. This ensures that the most relevant contextual excerpts are identified efficiently.

The retrieved excerpts, along with the processed query, are passed to the Google Gemini model. By combining contextual input with the query, Gemini generates a coherent, context-aware response tailored to the user's intent.

1) Data Logging: All interactions are logged for transparency and future analysis. Extracted materials and responses are stored in CSV files, enabling tracking of materials and user queries. Results are displayed in a user-friendly format through the Gradio interface. This approach, inspired by Retrieval-Augmented Generation (RAG), ensures efficient, scalable, and informative outputs.

The complete pipeline of our proposed RAG framework, from PDF ingestion and embedding to retrieval and final response generation, is shown in Figure 2.

IV. Simulation

A. Dataset Creation

We created a custom evaluation dataset using 15 peer-reviewed research papers in the domain of materials science. For each paper, 10 diverse queries were generated, totaling 150 queries. These were evenly distributed across four categories:

- Properties
- Synthesis Methods
- Applications
- Custom Queries

The custom queries included both numerical queries along with descriptive queries.

Numerical queries: Factual data such as dimensions, capacities, or values.

E.g.:

- *"Inter-layer distance in graphene"*
- *"Young's modulus of the material"*
- *"Li-storage capacity of germanium-based materials"*

Descriptive queries: Require understanding rankings, comparisons, or concepts.

E.g.:

- *"Which is the best semiconductor used for photocatalytic hydrogen generation?"*
- *"Which material exhibited the highest hardness?"*
- *"What is the matrix material?"*

As illustrated in Figure 4, the system allows users to submit custom queries that are processed across multiple uploaded PDFs. In this example, the query yields relevant results from the document *Reducing graphene oxide.pdf*, whereas the remaining two documents return no matches, demonstrating the system's ability to both identify pertinent information and acknowledge the absence of relevant content.

This design aimed to evaluate both factual extraction and conceptual understanding, mimicking realistic information needs.

B. Simulation Metrics

The model's performance was quantitatively assessed using the standard classification metrics based on a total of 150 queries. Each response of the query was annotated based on its correctness against the source paper. The annotation scheme follows standard parameters, defined below in accordance with our proposed methodology.:

- True Positive (TP): Answer is present in the source and correctly retrieved/generated.
- True Negative (TN): No answer exists and the model correctly returns none.
- False Positive (FP): Model produces an answer not supported by the source.
- False Negative (FN): A valid answer exists but was missed by the model.

Based on the values of TP, TN, FP, FN values, the following simulation metrics are calculated.

- **Accuracy**: The proportion of total predictions (both positive and negative) that were correct.
- **Precision**: The proportion of retrieved answers that were relevant (i.e., correct and supported by the source).
- **Recall**: The proportion of relevant answers (present in the source) that were successfully retrieved by the model.
- **F1-Score**: The harmonic mean of precision and recall, representing the overall effectiveness of the model. F1 score can be computed as:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

C. Human Cross Evaluation

To assess the factual correctness and relevance of the model-generated answers in the materials science domain, we conducted a structured human evaluation. This involved manually reviewing outputs from a total of 15 research papers, covering 150 queries evenly spread across four categories: properties, synthesis methods, applications, and custom questions (targeting numerical or conceptual clarity). For each query, the model's response was manually cross-checked against the corresponding source PDF to verify whether the answer was present and correctly aligned with the user's intent [5]. This ensured that the evaluation focused on both the accuracy and grounding of each generated answer.

TABLE I: Classification Metrics for Model-Generated Answers

Model	Accuracy	Precision	Recall	F1-Score
Matscibert	0.786	0.905	0.838	0.858
all-mini-lm	0.670	0.843	0.727	0.775

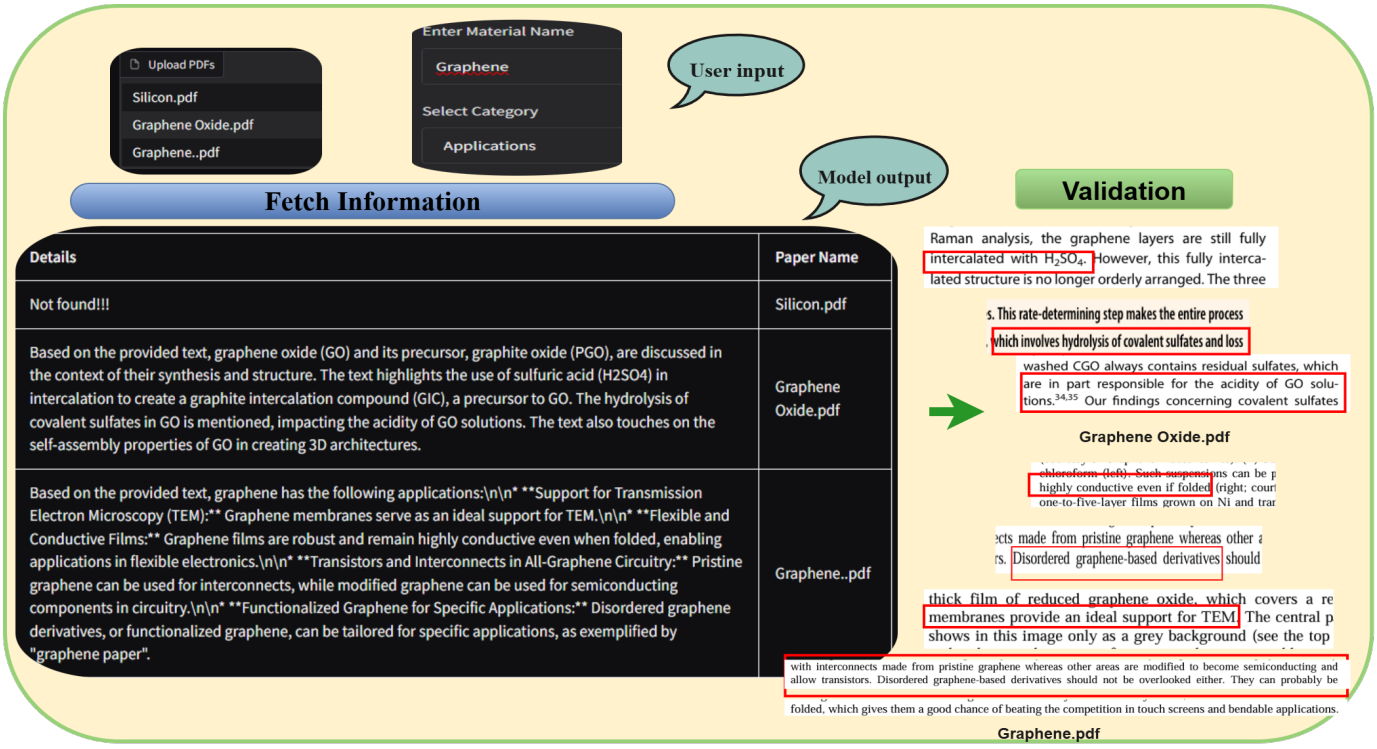

Fig. 3: Interface for viewing extracted applications of materials,with the validation of extracting the text from the pdf.

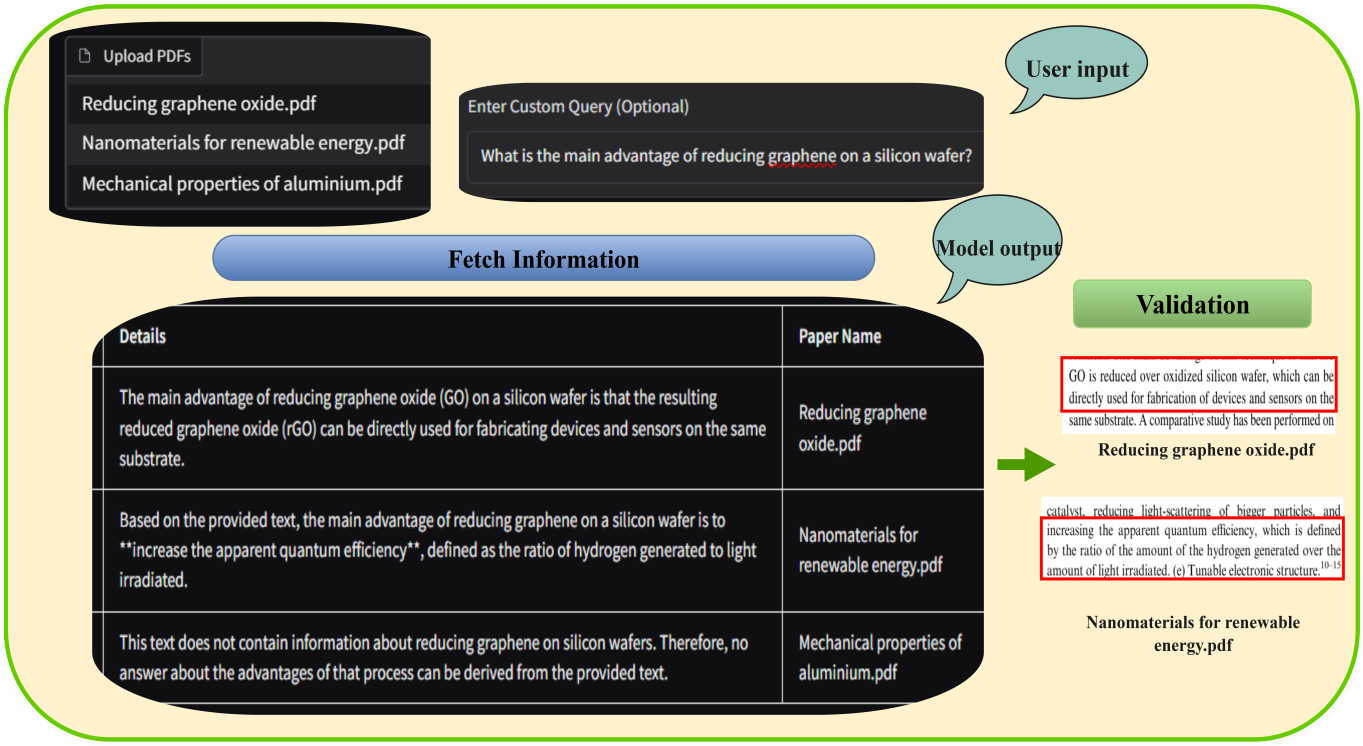

Fig. 4: User custom query is processed across three PDFs, retrieving relevant content from the first document only.

D. Examples

Figures 3 and 4 demonstrate how the system processes user queries across multiple PDFs and validates outputs directly against the source text. In Figure 3, the model successfully extracts and verifies applications of graphene from the uploaded

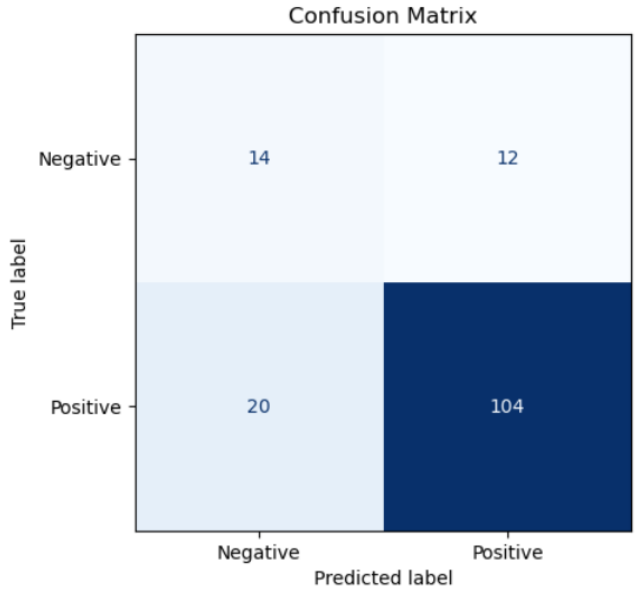

Fig. 5: Confusion matrix showing classification performance of the RAG framework using Matscibert model

documents, ensuring that each answer is explicitly grounded in the PDFs. Figure 4 further highlights the system's ability to handle custom queries, retrieving the correct explanation of reduced graphene oxide from the appropriate document while transparently indicating the absence of relevant content in others. Together, these examples illustrate the framework's precision, transparency, and reliability in contrast to generalized LLMs, which often provide unverified or hallucinated responses.

E. Why is it better than models like ChatGPT

While models like ChatGPT are powerful LLMs, it has clear limitations for domain-specific tasks like materials science information retrieval. A key drawback is that such models cannot process multiple PDFs at once. For example, uploading 15 research papers into ChatGPT simultaneously is not possible.

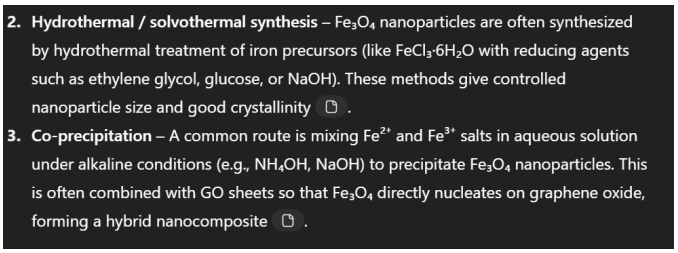

Fig. 6: Examples of hallucination in ChatGPT-like models

ChatGPT relies on its pre-trained knowledge, which may be incomplete, outdated, or irrelevant to the specific documents at hand. When queries are provided without attaching PDFs, ChatGPT often generates generic or incorrect answers because it attempts to predict a likely response rather than retrieve facts from the actual sources. Even when PDFs are supplied, ChatGPT may fabricate references or attribute information to a document where it does not actually exist. This behavior reduces reliability, especially for scientific research that demands precision and verifiable grounding in source material, as shown in Figure 6, where ChatGPT produces hallucinated content while the RAG system provides source-based synthesis methods.

In contrast, our framework is explicitly designed to retrieve answers only from the uploaded PDFs. By combining MatSciBERT embeddings, FAISS indexing, and Google Gemini API for generation of meaningful sentences, the system ensures responses are tightly coupled with the source documents. This avoids hallucinations and maintains transparency by clearly returning "Not found!!!" when relevant content is absent.

In short, while models like ChatGPT excels in open-ended reasoning, it cannot guarantee source-grounded, multi-document retrieval. Our RAG-based framework directly addresses this gap by ensuring factual accuracy, domain relevance, and scalability for handling large sets of research papers.

F. Results

The system efficiently extracts and analyzes material-related information from scientific PDFs using a combination of embeddings (MatSciBERT), FAISS-based semantic search, and Gemini for contextual generation. A Gradio interface supports PDF upload, query input, and result display.

1) Material and Information Extraction: Materials and chemicals are automatically extracted from PDFs using Gemini prompts. Users can submit queries, and Gemini generates concise responses based on the best matches from the documents.

2) Data Handling and Output: All results are structured as dictionaries and displayed as tables in Gradio Interface, with CSV export support for further usage.

3) Performance, Accuracy and Human Evaluation: Chunked PDF text (500 characters) ensures efficient vector search via FAISS. The use of MatSciBERT and Gemini boosts response accuracy, though output quality depends on input clarity. If context is missing, the system transparently returns "Not found!!!"

To assess the system's reliability, each response was manually checked for correctness, completeness, and contextual relevance. The system achieved an overall accuracy of 0.767, demonstrating a high degree of alignment with ground-truth information found in the documents.

In cases where the query lacked sufficient context or the information was absent from the PDF, the system transparently returned "Not found!!!", which reflects its ability to avoid hallucination and maintain response integrity.

Overall, the system offers fast, accurate, and scalable extraction and analysis of materials data, with potential for future enhancements in storage, UI, and analytics.

V. Conclusion

In this work, we proposed an effective method to mine material science based information out of PDFs through RAG framework. The system gathers related document pieces promptly and precisely by integrating the strength of KNN, MatSciBERT, and Google Gemini Model. This approach simplifies material science information to make it more readily available by enabling users to efficiently pull out specific material properties, applications, and synthesis processes from huge datasets. Although the system is very precise and scalable in processing large technical texts, processing time can impact its performance level when processing very large datasets.

Future attempts will concentrate on enhancing processing efficiency through additional data source linking, model refinement, and leveraging sophisticated embeddings to improve retrieval speed and accuracy. Further efforts will also be made to map the complex interrelation between applications and the characteristics of materials. This will include examining how certain material features affect their applicability for different purposes, fostering creation by more successfully matching material attributes with specific use cases.

References

[1] Ghanshyam Pilania. Machine learning in materials science: From explainable predictions to autonomous design. *Computational Materials Science*, 193:110360, 2021.

[2] Binglan Han, Teo Susnjak, and Anuradha Mathrani. Automating systematic literature reviews with retrieval-augmented generation: A comprehensive overview. *Applied Sciences*, 14(19):9103, 2024.

[3] Radeen Mostafa, Mirza Nihal Baig, Mashaekh Tausif Ehsan, and Jakir Hasan. G-rag: Knowledge expansion in material science. *arXiv preprint arXiv:2411.14592*, 2024.

[4] Nurshat Fateh Ali, Md Mahdi Mohtasim, Shakil Mosharrof, and T Gopi Krishna. Automated literature review using nlp techniques and llm-based retrieval-augmented generation. In *2024 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pages 1–6. IEEE, 2024.

[5] Juan José González Torres, Mihai Bogdan Bîndilă, Sebastiaan Hofstee, Daniel Szondy, Quang Hung Nguyen, Shenghui Wang, and Gwenn Englebienne. Automated question-answer generation for evaluating rag-based chatbots. In *1st Workshop on Patient-Oriented Language Processing, CL4Health 2024*, pages 204–214. European Language Resources Association (ELRA), 2024.