**Task - 2**

**1. Dynamic Query Expansion with Contextual Enrichment**

**Objective:**

Enhance the retrieval process by dynamically expanding the user's query with semantically similar terms or contextual keywords from the dataset, ensuring more accurate and diverse document retrieval.

**Implementation:**

* **Augment the Query**: Use a pre-trained language model (e.g., Sentence Transformer) to generate embeddings for the query and identify semantically related terms from the vector database.
* **Enrich the Retrieval Context**: Merge these terms with the original query to retrieve documents that are more contextually aligned with user intent.

**Benefits:**

* Ensures retrieval of more relevant documents by adding contextually related terms to the query.
* Improves the quality of generated answers by providing a richer context.

**2. Adaptive Retrieval with Feedback Loop**

**Objective:**

Optimize the retrieval and generation process by incorporating a feedback loop that ranks retrieved documents based on their relevance to user input and adjusts subsequent retrievals dynamically.

**Implementation:**

* **Scoring and Ranking**: Introduce a scoring mechanism to evaluate the relevance of retrieved documents.
* **Feedback Loop**: Use user feedback or model evaluation (e.g., cosine similarity of query and document embeddings) to refine the retrieval step.

**Benefits:**

* Improves the relevance of retrieved documents, leading to better answers.
* Enables adaptive refinement based on real-time evaluations of retrieval quality.

**Conclusion**

Both techniques—**Dynamic Query Expansion** and **Adaptive Retrieval with Feedback Loop**—work synergistically to optimize the RAG pipeline. While the former enriches the query for better initial retrieval, the latter ensures iterative improvement of document relevance. These methods together improve the precision and quality of generated answers, enhancing the user experience of the QA bot.