# RAG Types: Advantages, Disadvantages, Use Cases, and Additional Information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RAG Type | Advantages | Disadvantages | When to Use | Additional Information |
| Hybrid RAG | - High accuracy by combining multiple information sources - Handles diverse types of data (structured, unstructured) well - Robust in challenging scenarios | - Complexity in implementation - Higher computational resources required - Increased latency | - When accuracy is paramount, and there are multiple data types | Combines retrieval-based techniques (like search engines or databases) and generation-based techniques (like GPT-based models) to provide comprehensive responses. |
| Generative RAG | - Provides flexible and creative responses - Can generate human-like content - Capable of handling open-domain questions | - Risk of generating hallucinated information - Requires more extensive training data | - For open-ended or creative tasks, generating human-like answers | Focuses more on generative approaches by leveraging large language models (LLMs) to generate answers, which may include context-based information fetched from external sources. |
| Retrieval RAG | - Provides precise, contextually relevant information - Efficiently scales with large data - Suitable for factual accuracy | - Limited flexibility in response generation - May not adapt well to highly abstract or creative tasks | - When factual correctness is a priority | Primarily relies on robust retrieval systems such as search engines or database queries to provide pre-fetched content that is more factual and contextually accurate. |
| Knowledge-based RAG | - Integrates domain-specific knowledge - High reliability in specific domains - Supports real-time information and knowledge updates | - Limited to the knowledge base updates - May not perform well in general contexts | - When domain-specific knowledge is required | Uses a structured knowledge base to answer questions, making it ideal for use cases like medical consultations, technical troubleshooting, and specialized customer support. |
| End-to-End RAG | - Seamless integration of retrieval and generation - Provides a unified framework - Can be fine-tuned for specific applications | - Difficult to debug errors in responses - High model complexity | - When there is a need for streamlined, single-model architecture | These RAG systems combine both retrieval and generation processes in a tightly integrated end-to-end model, making them suitable for applications requiring high cohesion between information retrieval and generation. |