AI / ML Notes: RAG, LLMs, Agents & Vector Databases

These notes provide a concise academic overview of modern Generative AI concepts, designed for use in Retrieval-Augmented Generation (RAG) and Agentic AI systems.

1. Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) enhances Large Language Models by grounding responses in external knowledge sources. Relevant information is retrieved at query time and supplied to the model as context.

- Reduces hallucinations by grounding responses in real documents
- Supports private and domain-specific knowledge bases
- Uses embeddings and vector similarity search
- Commonly applied in enterprise chatbots and assistants

2. Large Language Models (LLMs)

Large Language Models are transformer-based neural networks trained on massive text datasets. They can understand context, generate coherent text, and follow instructions.

- Trained using self-supervised learning
- Leverage attention mechanisms for context awareness
- Can be instruction-tuned for better task performance
- Examples include GPT, FLAN-T5, and LLaMA

3. AI Agents and Agentic Systems

AI agents are systems capable of making decisions and taking actions to achieve goals. Agentic systems introduce reasoning, planning, and tool usage.

- Follow a perceive–reason–act loop
- Can decide when to retrieve information
- Improve efficiency by avoiding unnecessary operations
- Common in autonomous and semi-autonomous AI systems

4. Vector Databases

Vector databases store embeddings and enable semantic search by comparing vector similarity instead of exact keyword matches.

- Store high-dimensional numerical vectors
- Enable fast nearest-neighbor search
- Critical component of scalable RAG systems
- Examples: ChromaDB, FAISS, Pinecone, Weaviate

Conclusion: Modern AI systems combine LLMs, vector databases, and agentic reasoning to build intelligent, efficient, and reliable applications. Agentic RAG represents a key architecture in this evolution.