Embeddings – What & Why

• Embeddings = numerical vector representations of text/images/audio.
• Map meaning into high-dimensional space: similar concepts = close vectors.
• Essential for RAG: enable semantic search across enterprise KBs.

How Embeddings Work

- 1. Input text → tokenize → encoder neural net → dense vector.
- 2. Store vectors in vector DB.
- 3. Retrieve by similarity (cosine similarity).
- Diagram: Text → Encoder → Vector → Cluster in space.

Types of Embedding Models

- General-purpose (e.g., MiniLM, OpenAI text-embedding-3-small).
- Domain-specific (finance/legal tuned, e.g., FinBERT).
- Multimodal (CLIP, ImageBind) – map text ↔ images/audio.

Which to Use When?

- MiniLM → demos, fast prototypes.
- OpenAI text-embedding-3-large → production-grade precision.
- Domain-tuned (FinBERT) → compliance, contracts.
- Multimodal → receipts, video, audio transcripts.

Attendee Q&A

• Q: Why not just use 128k context?
• A: Embeddings cheaper, scale to millions, faster retrieval.
• Q: How big is an embedding vector?
• A: MiniLM=384 dims, OpenAI large=3072 dims.
• Q: Do embeddings handle synonyms? Yes, cluster semantically.
• Q: Multilingual? Use paraphrase-multilingual-MiniLM.