LLM Apps

The RAG technique

Remember: How to overcome the context window limits?

- Training an LLM from scratch with our data.
  - Hugely expensive. Impractical for most in reality.
- Adding our data to an already trained LLM (fine-tuning).
  - Very expensive and technically very complex. Impractical for most in reality.
- With the RAG (Retrieval-Augmented Generation) technique, we divide our data into small segments, allowing the LLM to use them within the limits of its context window.
  - This is the technique used today by virtually all LLM applications.

The RAG technique

- Preparations:
  - Divide your data into small segments.
  - Convert the small segments into numbers (“embeddings”).
  - Load the embeddings into a vector database.

- Now when you ask (“query”) the LLM:
  - The LLM goes to the vector database and searches (“indexing”) for data that only (“semantic similarity”) answers your question.

Therefore, when using RAG it is said that:

- The ability to speak (“language generation”) comes from the Foundation LLM.
- The specific knowledge (“knowledge representation”) comes from the vector database.
- In other words:
  - The Foundation LLM acts like a person who knows how to speak but is not familiar with your data.
  - The vector database acts as the expert knowledge that you add to that Foundation LLM to make it behave like a person who knows how to speak about your data.