Talk with your data: building a RAG system for searching in natural language

Enrico Zimuel, *Tech Lead & Principal Software Engineer*

Feb 26, 2025 - AI Festival, Milan (Italy)

Agenda

- Large Language Model (LLM)
- Limitation of LLMs
- Retrieval Augmented Generation (RAG)
- Embedding and Vector Search
- Langchain
- Ollama
- Elasticsearch
- RAG demo

Image generated using dall-e-3

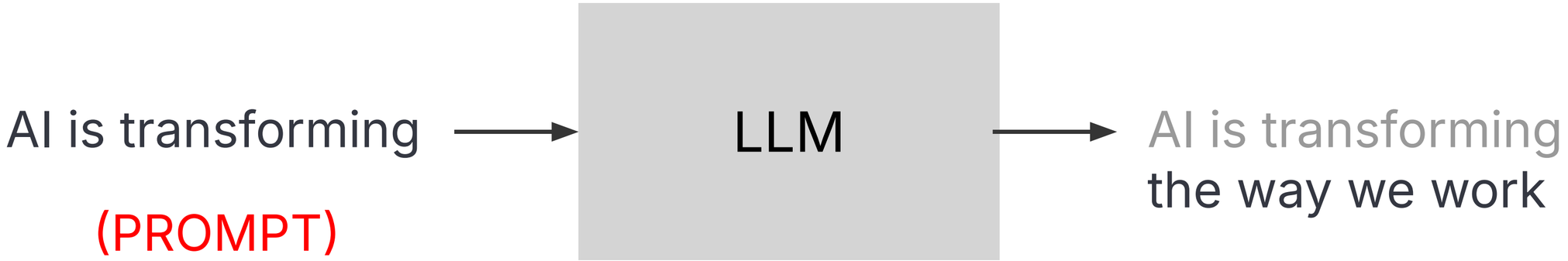
elastic

Large Language Model (LLM)

elastic

LLM

- **Large Language Model** (LLM) are probabilistic models that produce sentence in natural language
- These models work by completing sentences



elastic

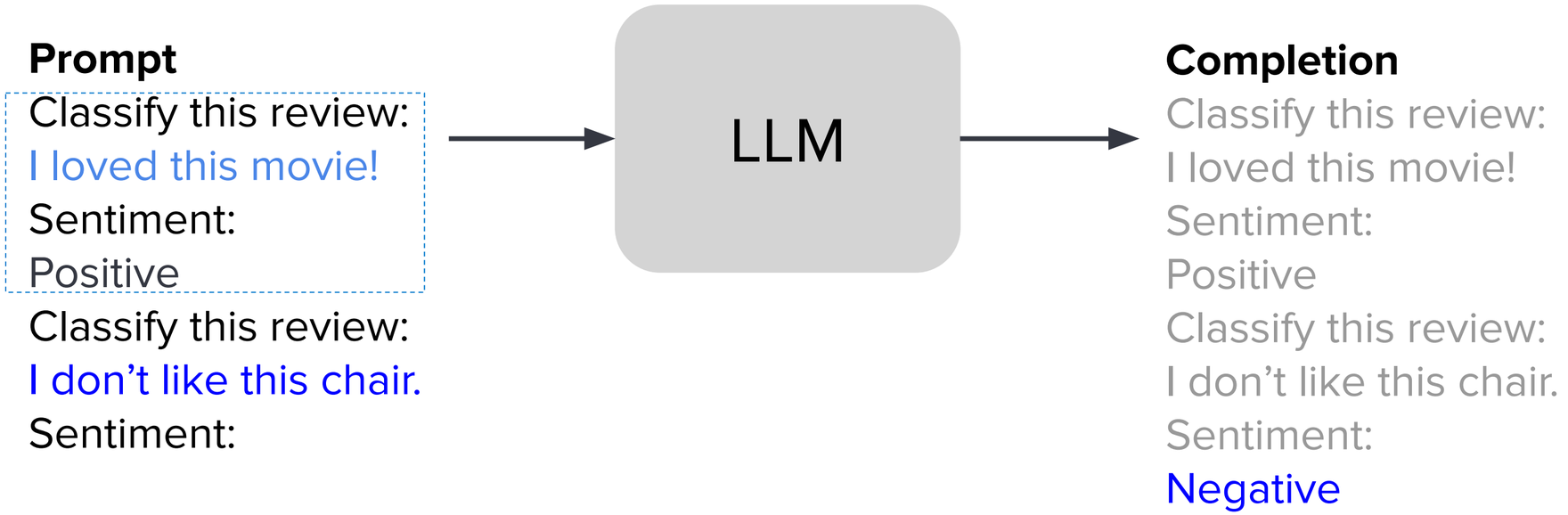
Prompt engineering

- You can encounter situations where the model doesn't produce the outcome that you want on the first try
- You may have to revisit the language several times to get a good answer
- The development and improvement of the prompt is known as **prompt engineering**
- One powerful strategy is to include examples of the task that you want the model to carry out inside the prompt
- This is called **In-Context Learning (ICL)**

ICL - zero shot inference

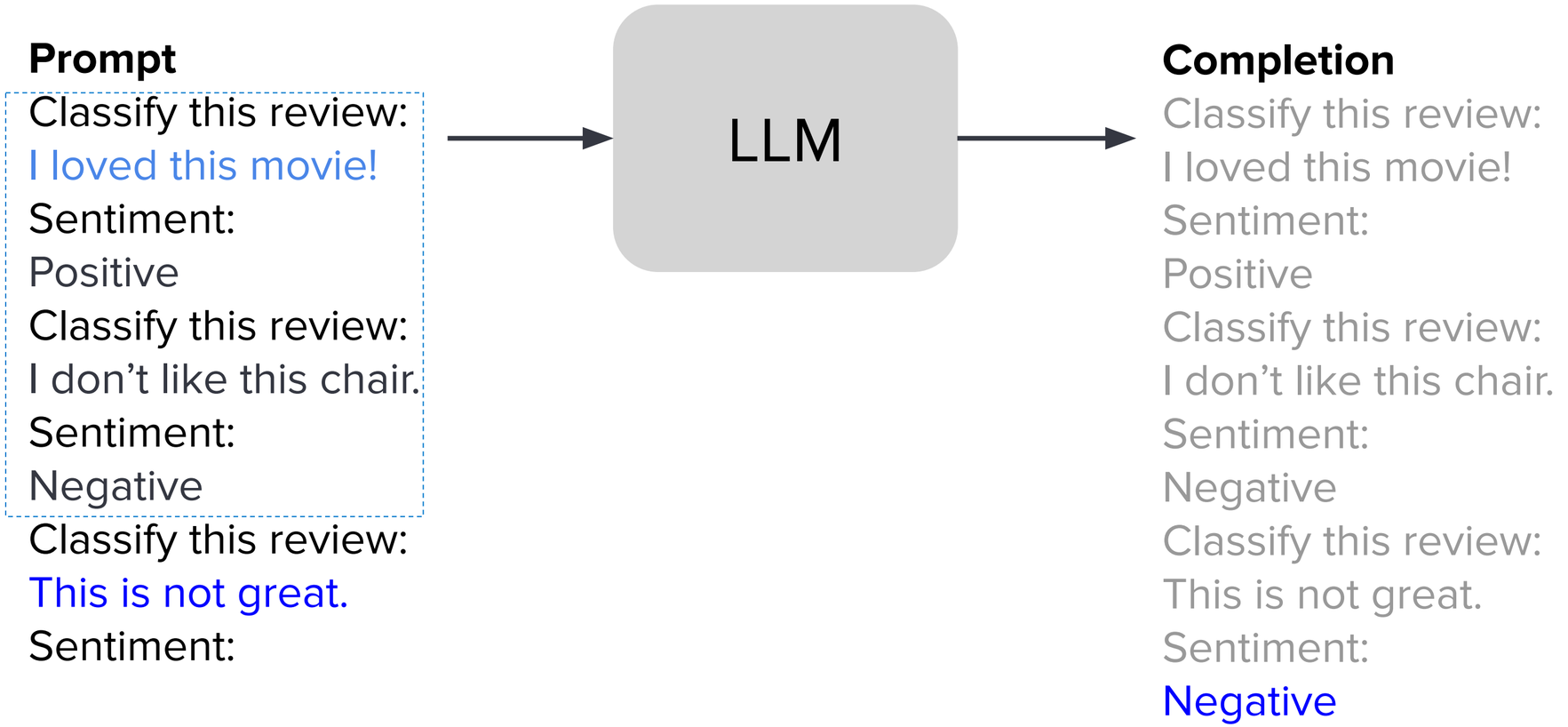
elastic

ICL - one shot inference



elastic

ICL - few shot inference



elastic

LLM limitations

- **Prone to Hallucinations**: Since an LLM is a probabilistic model, it can generate incorrect or nonsensical information
- **No sources:** The output of an LLM does not provide sources for its information (again hallucinations)
- **Fixed Knowledge**: The model's knowledge is static, meaning it does not learn or adapt from interactions
- **Difficult to Update**: Expanding an LLM’s knowledge requires retraining or fine-tuning, which is complex, resource-intensive, and time-consuming

Retrieval-Augmented Generation (RAG)

elastic

Retrieval-Augmented Generation (RAG)

- **RAG** is a technique in natural language processing that combines information retrieval systems with **Large Language Models** (LLM) to generate more informed and accurate responses
- It is composed by the following parts:
  - **Retrieval-Augmented**
  - **Generation**

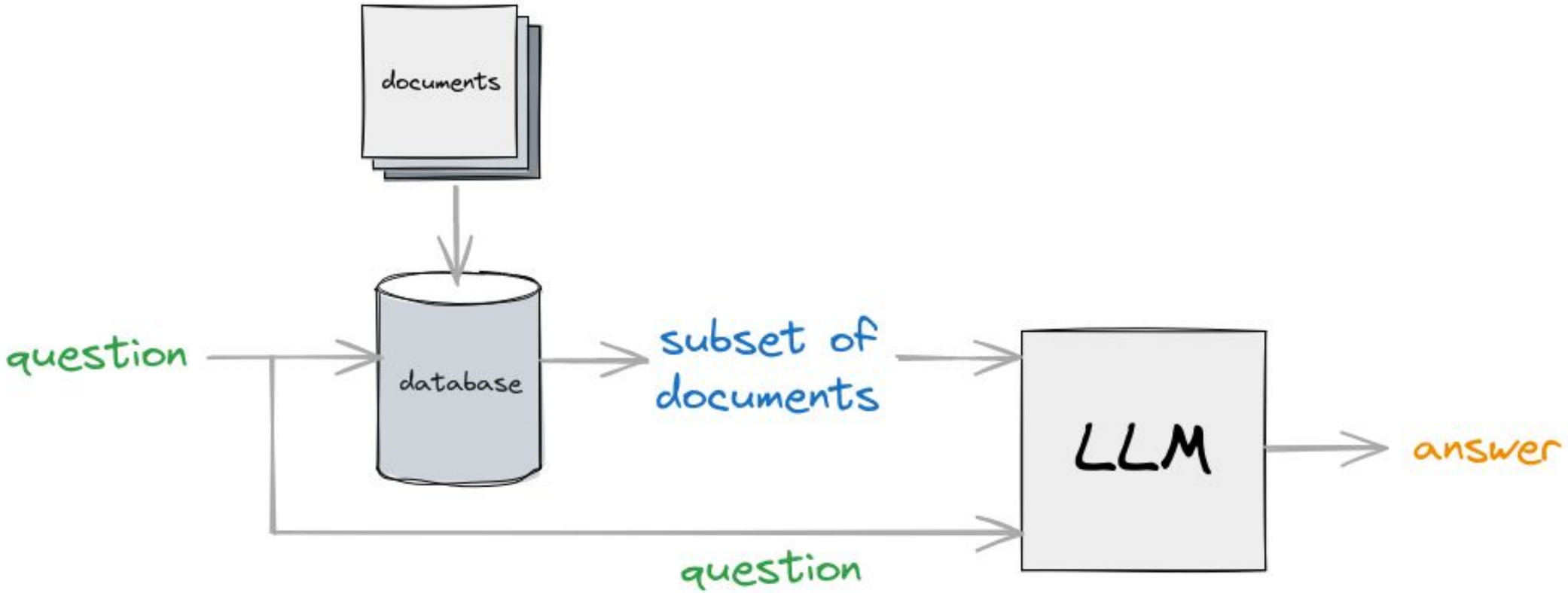
Generation

- LLMs are very powerful but have some limitations:
  - **No source** (potential hallucinations)
    - How can I verify the information coming from an LLM?
    - What sources has been used to generate the answer?
  - **Out of date**
    - An LLM is trained in a period of time
    - For update we need to retraining the model (very expensive)

Retrieval-Augmented

- We collect sets of private or public document
- We build a **retrieval system** (e.g. a database) to extract a subset of documents using a **question**
- Then we pass the **question + documents found** to an LLM as prompt with a context
- The LLM can give an answer using the updated documents

RAG architecture



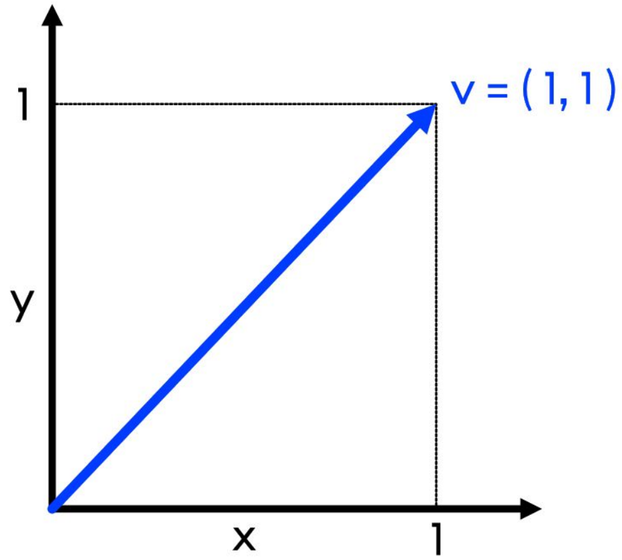
elastic

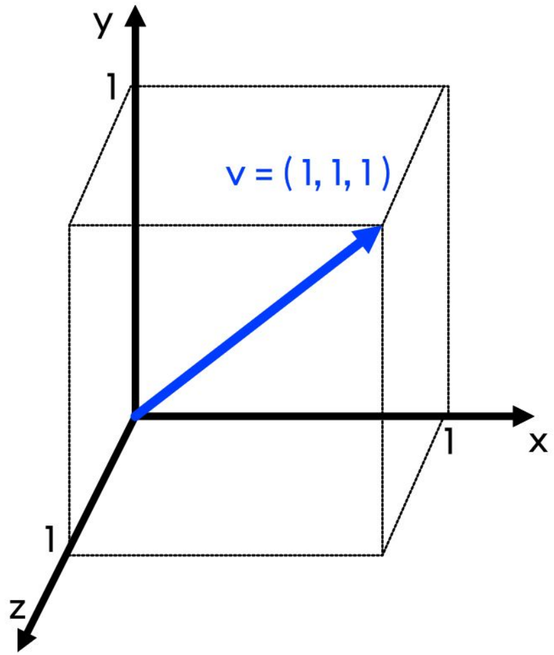
Retrieve documents from a question

- How we can retrieve documents in a database using a question?
- We need to use **semantic search**
- One solution is to use a **vector database**
- A vector database is a system that uses **vectors** (set of numbers) to retrieve information

What is a vector?

- A vector is a set of numbers
- Example: a vector of 3 elements [2, 5, -10]
- A vector can be represented in a multi-dimensional space (eg. Llama3.2 uses 3072 dimensions)

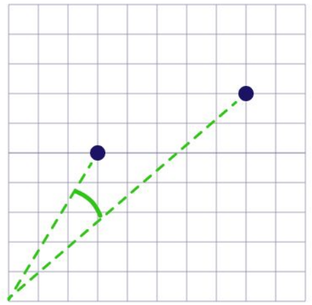




elastic

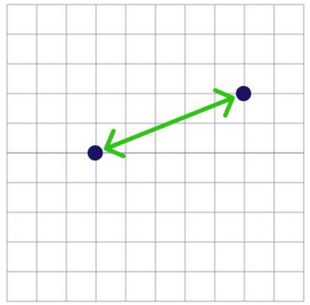
Similarity between two vectors

- Two vectors are (semantically) similar if they are close to each other
- We need to define a way to measure the similarity



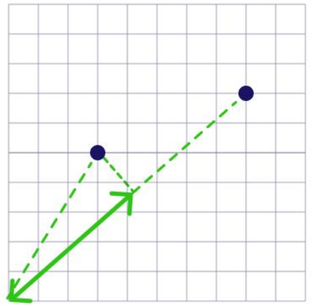
Cosine Distance

$$1 - \frac{A \cdot B}{||A|| \quad ||B||}$$



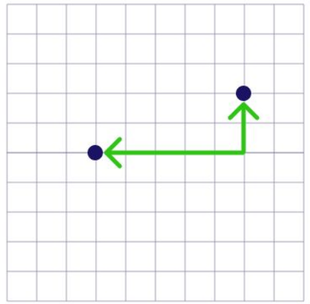
Squared Euclidean (L2 Squared)

$$\sum_{i=1}^{n} (x_i - y_i)^2$$



Dot Product

$$A \cdot B = \sum_{i=1}^{n} A_i B_i$$



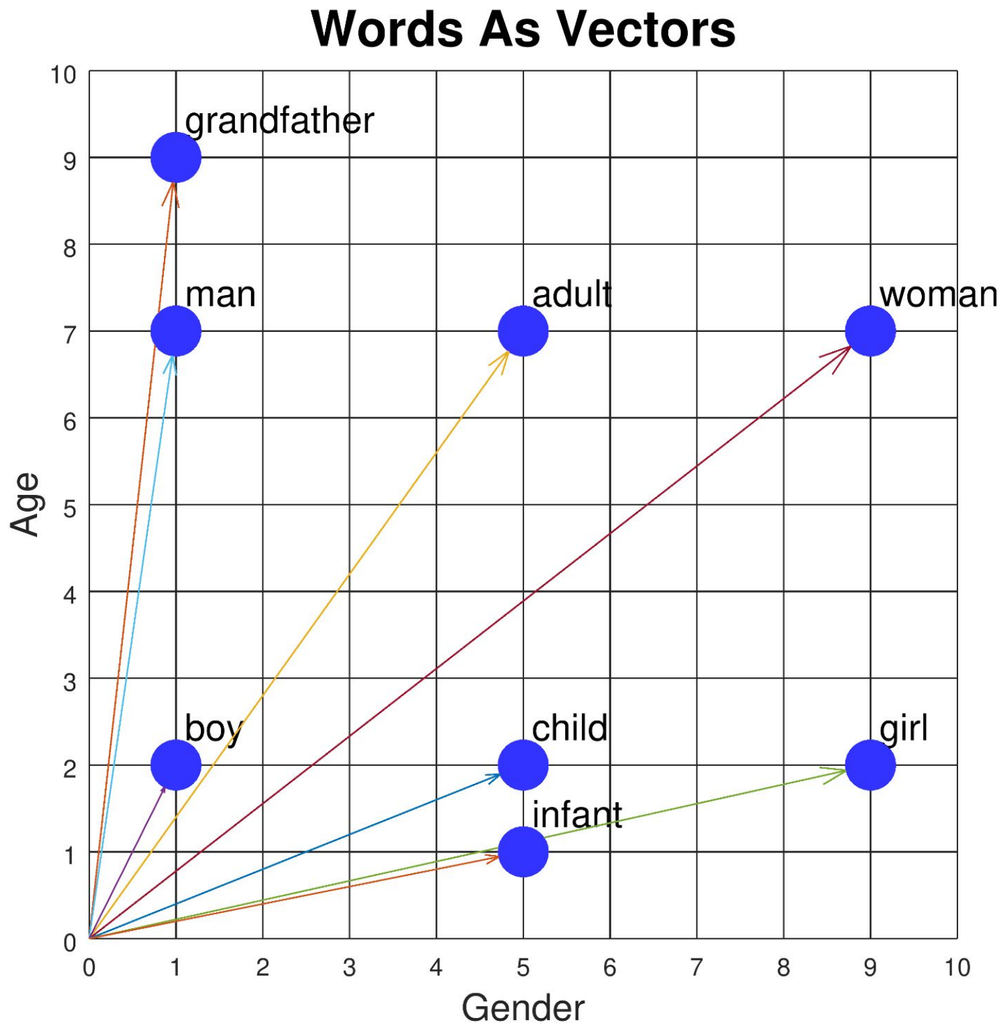
Manhattan (L1)

$$\sum_{i=1}^{n} |x_i - y_i|$$

elastic

Embedding

- Embedding is the translation of an input (document, image, sound, movie, etc) to a vector
- There are many techniques, using an LLM typically this is done by a neural network
- The goal is to group information that are semantically related to each other
- https://projector.tensorflow.org/



elastic

Vector database + LLM

- The search query (***question***) is in natural language
- We use semantic search to retrieve top-n relevant documents (***context***)
- We send the following prompt to the LLM (example):
  - *Given the following **{context}** answer to the following **{question}***

elastic

Split the documents in chunk

- We need to store data in the vector database using chunk of information
- We cannot use big documents since we need to pass it in the context part of the prompt for an LLM that typically has a token limit (e.g. Llama3.2 up to 128K)
- We need to split the documents in **chunk** (part of words)

RAG demo:

LangChain + Ollama + Elasticsearch

elastic

LangChain

- LangChain is an open source composable framework to build with LLMs
- Supports all the LLMs (see here)
- Integrations with many vector databases (e.g. Chroma, Elasticsearch, Milvus, Qdrant, Redis)
- Available for Python (98K ⭐) and Javascript (13K ⭐)
- MIT license
- Other interesting projects: LangGraph (MIT license) and LangSmith (commercial)

elastic

Ollama

- Ollama is a software for downloading and running LLMs locally
- Llama 3, Phi 3, Mistral, Gemma, and other models
- Simple command line tool:
  - ollama pull llama3.2:3b
  - ollama run llama3.2:3b

elastic

Elasticsearch (vector database)

- Elasticsearch is Free and Open Source (AGPL), Distributed, RESTful Search Engine
- Distributed search and analytics engine, scalable data store and **vector database** optimized for speed and relevance on production-scale workloads
- You can run it locally with a single command:
  - **curl -fsSL https://elastic.co/start-local | sh**

elastic

DEMO

https://ela.st/langchain-llama-elasticsearch

elastic

References

- What is retrieval-augmented generation? IBM research
- Ashish Vaswan et al., Attention Is All You Need, Proceedings of 31st Conference on Neural Information Processing Systems (NIPS 2017)
- Albert Ziegler, John Berryman, A developer's guide to prompt engineering and LLMs, Github blog post
- Sebastian Raschka, Build a Large Language Model (From Scratch), Manning, 2024
- Elasticsearch as vector database, Elastic Search Labs
- Elasticsearch search relevance, Elastic Search Labs
- E.Zimuel, Retrieval-Augmented Generation for talking with your private data using LLM, AI Heroes 2023 conference, Turin (Italy)

elastic

Thanks!

More information: www.elastic.co

Contact information: enrico.zimuel (at) elastic.co

elastic