Customizing a Chatbot Using Embeddings Pt.1

Theoretical Prerequisites

Iverina Ivanova

Goethe University Frankfurt a.M.

Dec. 11th, 2023

Plan

- Why customizing a chatbot?
- Aim
- Text embeddings
- Applications of embeddings
- Hugging Face

Why customizing a chatbot?

- A different use case of applications powered by large language models (LLMs) (a developer's perspective)
- A domain-specific/purpose-specific chatbot which interacts with user-defined texts (e.g., linguistic papers, the course syllabus, etc.)

Aim

- To build a basic chatbot that takes raw unstructured data as input and turns it into a Question-Answer (Q-A) chain.
- For this purpose, I used:
  - Sentence transformers – a framework that computes sentence/text embeddings
  - Faiss – a Python library used for similarity searches
  - Langchain – a framework for developing applications powered by language models
  - falcon-7b-instruct – an open-source LLM

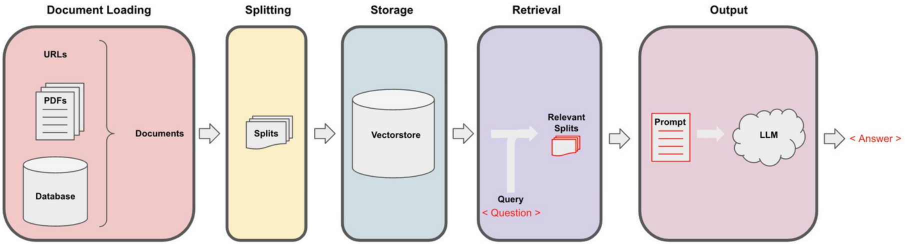

Figure 1: LLM-based Q-A System Pipeline (Source: Langchain)

In a nutshell...

The use of embeddings is the underlying method that translates human speech into a list of abstract numeric representations which encode context-aware information.

What is an embedding?

- An embedding is a way of representing data (text, audio, image) as points in a vector space where the locations are semantically meaningful.
- Embeddings are numeric representations of data (floating point numbers of the form [0.029, 0.112, 0.898, ..., 0.236]). Each embedded object – a word, a sentence or a document – is represented as a point in a vector space. (Namjoshi & Ng 2023)

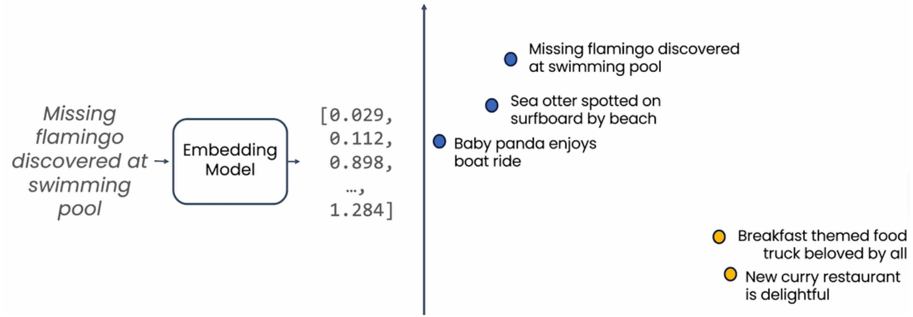
Missing flamingo discovered at swimming pool
Embedding Model
[0.029, 0.112, 0.898, ..., 1.284]
Missing flamingo discovered at swimming pool
Sea otter spotted on surfboard by beach
Baby panda enjoys boat ride
Breakfast themed food truck beloved by all
New curry restaurant is delightful

Figure 2: Embedding process (Namjoshi & Ng 2023)

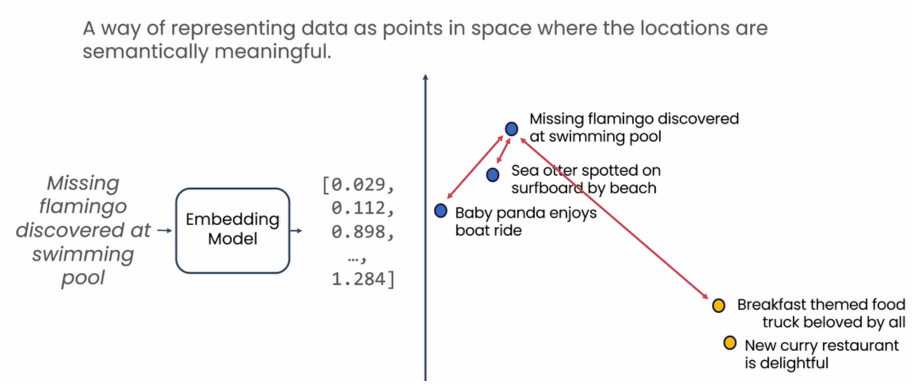
A way of representing data as points in space where the locations are semantically meaningful.
Missing flamingo discovered at swimming pool
Embedding Model
[0.029, 0.112, 0.898, …, 1.284]
Missing flamingo discovered at swimming pool
Sea otter spotted on surfboard by beach
Baby panda enjoys boat ride
Breakfast themed food truck beloved by all
New curry restaurant is delightful

Figure 3: Embeddings and semantic similarity (Namjoshi & Ng 2023)

What is an embedding?

- Each embedding has a dimension (length) which encodes different aspect of the object's properties (e.g., the context in which a word is used; the word order; the words that co-occur with it in the sentence, etc.) Thus, the embedding is contextually informed.
- An embedding with 1536 dimensions means that each object is represented by a vector with 1536 elements.
- High-dimensional embeddings may capture more information, but may require more processing capacity and storage; may also capture noise and lead to overfitting. (Vaj 2023)
- Reducing the dimensionality of an embedding might lead to information loss, but it helps us focus on the most important aspects of the object's properties and result in models are less prone to overfitting. (Vaj 2023)

Traditional embedding methods

- Embed each word separately and compute the mean of all word embeddings to get the sentence-level embedding
- Drawback: The sentence-level embeddings ignore word order and do not consider the rest of the words that appear in the sentence. Thus, sentences containing the same lexical units
  e.g., *"John kicked Jane" and "Jane kicked John"*
  will have identical embeddings and will be rendered wrongly as semantically similar.

Modern embedding methods

- Make use of a transformer neural network to compute a context-aware representation of each word in the sentence. (Namjoshi & Ng 2023)
- Embed each token(subword) by taking into account contextual properties.
- The average of the context-aware representations is used for the sentence embeddings. This makes embeddings algorithms more sophisticated and thus sentences containing the same lexical units but have different meanings will not have the same embeddings.

How do text embeddings work?

A short intro to text embeddings by Andrew Ng (DeepLearning.AI)

Applications of embeddings

- text classification
- semantic search (to find the embedding values that are closest to the query embedding)
- clustering
- recommendation systems

Sentence similarity search

- identifying which sentences are similar and which different in meaning by comparing the embeddings values of sentences
- cosine similarity metrics – computes the similarity between embeddings of input sentences/texts
  Source: Sentence Similarity

The Falcon 7B Instruct model

Falcons are not what they used to be

The Falcon 7B Instruct LLM model

- accessible from Hugging Face
- open-source (creating a token doesn't cost a thing)
- it requires at least 16GB of memory (it can run on Google Colab)
- instruction-based large language model – i.e., fine-tuned on Q-A/instruction datasets

Hugging Face

- Hugging Face - a major hub for machine learning (ML) which gives access to:
  - pre-trained, open-source LLMs,
  - dataset repository,
  - spaces (which is a platform that allows developers to build and deploy their LLM-based apps).

Hugging Face Transformers

- a Python library that facilitates the downloading and training of ML models easy
- originally developed for NLP tasks – question-answering, summarization, topic modeling, etc.
- its current function expands into other domains – computer vision, audio processing, etc.
- it allows within a few lines of code to take a raw text, convert it into numerical representations, pass it into the model, and translate the numerical output back into human speech.

References

- Ankit, Utkarsh. 2020. *Transformer Neural Network: Step-By-Step Breakdown of the Beast*. Accessible from: Towards Data Science
- Espejal, Omar. 2022. *Getting Started With Embeddings*. Accessible from: HuggingFace
- Mohiuddin, Ahmed. 2023. *Using AI to chat with documents: Leveraging LangChain, FAISS, and OpenAI*. Accessible from: medium.com
- Namjoshi, Nikita & Andrew Ng. 2023. Understanding and Applying Text Embeddings. Accessible from: Deep Learning.AI
- Reimers, Nils. 2022. *SentenceTransformers Documentation.* Accessible from: Sentence-Transformers
- Sahrane, Selim. 2023. *Question Answering over text files with Falcon 7B and LangChain*. Accessible from medium.com

References

Vaj, Tiya. 2023. *The Relationship Between High Dimensionality and Overfitting.* Accessible from: medium.com

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser & Illia Polosukhin. 2023. *Attention Is All You Need.* https://arxiv.org/abs/1706.03762v7