README

Vector Databases are optimized for storing and performing operations on large amounts of vector data, often processing hundreds of millions of vectors per query, and doing it significantly faster than traditional databases are able to. In recent times, Vector Databases have been gaining popularity at an alarming rate due to their synergies with AI models like ChatGPT. We are using the free tier currently.

Pinecone makes it easy to provide long-term memory for high-performance AI applications. It’s a managed, cloud-native vector database with a simple API and no infrastructure hassles. Pinecone serves fresh, filtered query results with low latency at the scale of billions of vectors.

LangChain:

LangChain is a framework for developing applications powered by language models. It enables applications that:

Are context-aware: connect a language model to sources of context (prompt instructions, few shot examples, content to ground its response in, etc.)
Reason: rely on a language model to reason (about how to answer based on provided context, what actions to take, etc.)

At it’s very basic, it looks something like this:

```
chain = prompt | model | output_parser
```

The | is like a Unix pipeline operator, piecing together the prompt, with the model and the output parser.

For an RAG-Retrieval Augmented Generation, it will be slightly different looking like:

```
chain = (
{"context": retriever, "question": RunnablePassthrough()}
| prompt
| model
| StrOutputParser()
)
```