Learn Vector Database using Python, Pinecone, LangChain, Open AI, Hugging Face and build out AI, ML , Chat applications

**What you'll learn**

* Pinecone Vector Database, LangChain, Transformer Models for vector embedding, Generative AI, Open AI API Usage, Hugging Face Models
* Master the essential techniques for vector data embedding, indexing, and retrieval.
* A Practical Code Along with Semantic Search Use Case in Detail with Named Entity Recognition
* Developing an AI Chat Bot for Cognitive Search on Private Data Using LangChain
* Understand the fundamentals of vector databases and their role in AI, generative AI, and LLM (Language Model Models).
* Explore various vector database technologies, including Pinecone, and learn how to set up and configure a vector database environment.
* Learn how vector databases enhance AI workflows by enabling efficient similarity search and nearest neighbor retrieval.
* Gain practical knowledge on integrating vector databases with Python, utilizing popular libraries like NumPy, Pandas, and scikit-learn.
* Implement code along exercises to build and optimize vector indexing systems for real-world applications.
* Explore practical use cases of vector databases in AI, generative AI, and LLM, such as recommendation systems, content generation, and language translation.
* Understand how vector databases can handle large-scale datasets and support real-time inference.
* Gain insights into performance optimization techniques, scalability considerations, and best practices for vector database implementation.

Goal of this assignment is to make yourself familiar how to use the hugging face documentation. In this particular assignment I will encourage you to go over this URL, read, understand and explore by yourself other models. <https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2>

Named Entity Recognition (NER)– proper names, organisations, places, location, dates, brands, product names, amounts of money as opposed to general nouns and verbs

Python module ‘spacy’ can be used to recognize named entities.

Use Cases:

Information extraction

Improved search and recommendation systems

Entity linking and knowledge graphs

Text summarisation and document organisation

Named entity disambiguation

Sentiment Analysis and opinion mining

Legal and Conpliance applications

Social media monitoring

Implementation

START -> Text Processing -> Part of Speech tagging (POS) -> Named Entity Recognition -> Entity Classification -> Post Processing -> Entity Disambiguation -> Output Generation -> END