In this project, demonstrate skills in **NLP**, **big data analysis**, and the creation of a **LangChain application** leveraging **vector databases**.

#### **Task Overview:**

You will be provided with a [**news article dataset**](https://docs.google.com/spreadsheets/d/1--c9SXNE-JboSu51fFQn8tZk-mPxfr9S/edit?usp=sharing&ouid=103976931801368455461&rtpof=true&sd=true) that includes a variety of articles spanning multiple domains (such as technology, finance, healthcare, etc.). Your goal is to:

1. **Analyze the Dataset**:
   * Perform **entity extraction** using an NLP model (such as SpaCy, Hugging Face, etc.) to identify company names mentioned in each article.
   * Categorize these companies by the **domain** or industry they are related to (e.g., Apple in technology, Pfizer in healthcare).
2. **Big Data Analysis**:
   * Use **Pandas** for processing the data and performing statistical analysis.
   * Analyze and visualize the **number of companies** mentioned in each domain and how frequently they appear.
3. **Build a LangChain Application**:
   * Use **LangChain** to build a simple application that leverages the dataset and a **vector database** (such as Pinecone, Weaviate, or FAISS) to store company-related embeddings.
   * Implement a functionality that allows users to **query** the application by company names and retrieve related articles or statistics from the dataset.

#### **Detailed Steps:**

### **1. Dataset Analysis:**

* **Entity Extraction**:
  + Use a pre-trained NLP model (like SpaCy or Hugging Face transformers) to extract **company names** mentioned in the dataset.
  + If necessary, fine-tune or customize the model to improve accuracy.
* **Categorization**:
  + For each extracted company, determine its **domain** (technology, healthcare, etc.). You can use predefined lists of companies, APIs, or manual categorization if needed.
* **Data Wrangling**:
  + Use **Pandas** to clean and preprocess the dataset, ensuring consistent formatting.
  + Count the occurrences of companies in different domains and across various articles.
* **Data Visualization**:
  + Create insightful **visualizations** (using Matplotlib/Seaborn) to show:
    - The **distribution** of companies across different domains.
    - **Frequency analysis** of company mentions in different sectors over time or across articles.

### **2. LangChain Application:**

* **Vector Database**:
  + Store company names or related entities as **vectors** using a vector database like Pinecone, Weaviate, or FAISS.
  + Encode the company names or their articles into embeddings using a model from Hugging Face or OpenAI’s API.
* **Query Functionality**:
  + Implement a **search** functionality in LangChain where users can input a company name and receive:
    - Relevant articles mentioning that company.
    - Statistics about how frequently the company appears in the dataset.
* **Optional Feature**:
  + Add a feature where users can search for companies by domain (e.g., "Show me all companies in the technology sector").

### **Deliverables:**

1. **Demo**:
   * A short video demonstrating the LangChain application and walking through your analysis.

### **Evaluation Criteria:**

* **Accuracy**: Correctness in identifying and categorizing companies.
* **Completeness**: Meeting all outlined tasks and functionality.
* **Innovation**: Any creative or additional features beyond the basic requirements.
* **Presentation**: Clarity and organization of the demo.

#### **Tools You Can Use:**

* **Pandas** for data processing and analysis.
* **NLP libraries** like SpaCy, Hugging Face, or NLTK for entity extraction.
* **Matplotlib/Seaborn** for data visualization.
* **LangChain** for building the application.
* **Vector DBs**: Pinecone, FAISS, or Weaviate for storing and querying embeddings.

#### **Submission:**

Please submit your assignment within **7 days** of receiving this prompt. If you have any questions, feel free to reach out to the Textify team.

Good luck, and we look forward to seeing your work!