**Brief Report**

**Data Science Project on Web Scraping, Entity Extraction, and Sentiment Analysis**

**Objective:**

The primary objective of this project is to build a system that scrapes news articles from URLs, extracts relevant entities (persons or organizations), and performs sentiment analysis on the content. The goal is to evaluate the performance of named entity recognition (NER) and sentiment analysis models on news articles.

**Approach:**

1. **Web Scraping:** To retrieve article content, I used the requests library to fetch the HTML of the provided URL. Since the articles often contain embedded media like JavaScript, a simple requests-based scraping method was insufficient. Therefore, I incorporated BeautifulSoup (BS4) for parsing the HTML and extracting relevant sections of the text. This approach enabled us to extract clean article text without the noise of JavaScript, advertisements, and other irrelevant content.
2. **Entity Extraction:** For entity extraction, I utilized spaCy, a widely used Natural Language Processing (NLP) library. SpaCy’s pre-trained model, en\_core\_web\_sm, was employed to identify named entities in the text. The entities were classified as either "PERSON" (for people) or "ORG" (for organizations), based on the spaCy model’s recognition. The extraction process included filtering out irrelevant words, retaining only those entities that were pertinent to the project.
3. **Sentiment Analysis:** Sentiment analysis was performed using the TextBlob library. This library provides a simple interface for processing textual data, including sentiment classification based on polarity scores. Articles with a positive sentiment polarity score were classified as "Positive", those with a negative score as "Negative", and articles with neutral polarity scores were classified as "Neutral". This method provided a straightforward way to classify sentiment in a way that aligns with common practices in text mining.
4. **Storage:** The data, including the URL, extracted entities, and sentiment analysis results, were stored in a local SQLite database. A table was created to save these results, ensuring the data could be retrieved easily for further analysis or reporting.
5. **Deployment:** The entire application was containerized using Docker to ensure that it could run seamlessly across different systems without dependencies or configuration issues. Docker Compose was used to orchestrate the deployment of services, including the web application and the database.

**Challenges Faced:**

1. **Entity Extraction Accuracy:**
   * The accuracy of entity extraction was affected by the nature of the text. While spaCy’s pre-trained model works well for most formal and structured content (such as news articles), it sometimes misidentified common nouns as entities. For example, in one of the test cases, "Himalayan" and "Trishul" were incorrectly identified as persons or organizations.
   * Some entities, especially geographical locations or lesser-known organizations, were missed by the model. A more specialized model or fine-tuning might improve the accuracy for specific domains, such as geographic or religious references.
2. **Sentiment Analysis Accuracy:**
   * The sentiment analysis using TextBlob provided decent results for straightforward and opinionated text. However, it struggled with more neutral content, classifying it incorrectly at times. Sentiment can vary greatly based on context, and the model does not account for this complexity well. It tends to give neutral or slightly positive results for factual or informational text, even if the tone of the article is not strongly positive.
   * For example, when analyzing an article about a natural landmark, the sentiment classifier might return a positive result simply due to the use of positive words like "beautiful," even though the overall sentiment might be neutral or factual.
3. **Database Integration and Deployment:**
   * Setting up the SQLite database was straightforward, but integrating the system with Docker posed some challenges, especially when ensuring that the local environment worked without errors. It took time to debug issues related to Dockerfile configuration and database connections across containers. However, once configured, Docker provided a stable environment for testing and deployment.

**Reflections on Accuracy:**

* **Entity Extraction:**
  + The performance of spaCy in identifying organizations and persons was generally good, but not flawless. In articles with complex structures or mixed contexts (like those involving multiple topics or locations), the model struggled to identify the correct entities. Custom-trained models or using a more domain-specific model could address this issue.
* **Sentiment Analysis:**
  + TextBlob proved to be an efficient tool for sentiment analysis, but its limitations became apparent with more nuanced content. For example, articles that express neutral or mixed sentiments were often misclassified. To enhance sentiment analysis, other approaches like using transformers-based models (BERT, RoBERTa) or fine-tuning TextBlob on a domain-specific dataset could improve accuracy.

**Conclusion:**

This project demonstrates the practical implementation of scraping, entity extraction, and sentiment analysis techniques to process news articles. While the models used provide reasonable results, there is room for improvement, particularly in handling complex or domain-specific text. The challenges faced in scraping, entity extraction, and sentiment analysis were mitigated through iterative improvements to the codebase and the incorporation of advanced NLP techniques. Further improvements could include using specialized models, increasing the robustness of web scraping, and experimenting with more advanced sentiment analysis techniques.