10 Academy: Artificial Intelligence Mastery

Week 7 Challenge Document

Date: 29 Feb - 04 Feb 2024

Building a Data Warehouse to store data on Ethiopian medical business data scraped from telegram channels

Overview

Business Need

You are a data engineer at **Kara Solutions,** a leading data science company with over 50+ data-centric solutions. You are tasked with Building a data warehouse to store data on Ethiopian medical businesses scrapped from the web and telegram channels. This project involves several key steps and considerations to ensure the data warehouse is robust, scalable, and capable of handling the unique challenges associated with scraping and data collection from Telegram channels. Additionally, it involves integrating object detection capabilities using YOLO (You Only Look Once) to enhance data analysis.

A data warehouse significantly enhances data analysis. With all data stored centrally, your team can perform comprehensive analyses to find valuable insights about Ethiopian medical businesses. This data helps identify trends, patterns, and correlations that are hard to detect with fragmented data, leading to better decision-making. A well-designed data warehouse also makes querying and reporting more efficient, enabling businesses to get actionable intelligence quickly and accurately.

The implementation of **ETL (Extract, Transform, Load)** and **ELT (Extract, Load, Transform)** frameworks is a key part of this setup. **ETL** processes involve extracting data from various sources, transforming it into a suitable format, and then loading it into the data warehouse. This ensures the data is clean, consistent, and ready for analysis. **ELT**, on the other hand, loads raw data into the data warehouse first and then transforms it as needed. This approach can be more flexible and efficient, especially with the processing power of modern data warehouses. These frameworks are crucial for keeping the data intact and usable, allowing seamless integration and transformation of different data sets.

Your job is to build a product that does the following

1. Develop data scraping and collection pipeline.
2. Develop data cleaning and transformation pipeline.
3. Object detection using YOLO.
4. Data warehouse design and implementation
5. Data integration and enrichment

Data and Features

You will be building a data warehouse for this week.

Learning Outcomes

Skills:

- Telegram scraping using Beautiful Soup, Scrapy, and Selenium
- Object detection using YOLO (You Only Look Once)
- Data cleaning and transformation using ETL processes
- Database schema design for data warehouses
- Implementing and configuring relational DBMS (e.g., PostgreSQL)
- Data integration and enrichment techniques
- End-to-end data pipeline development
- Testing and validation of data systems
- Deployment and maintenance of data warehouses

Knowledge:

- Identifying relevant data sources for Ethiopian medical businesses
- Principles of object detection and its applications
- Best practices in data cleaning and preprocessing
- Structuring data for efficient storage and retrieval in data warehouses
- Techniques for integrating and enriching data from multiple sources
- Implementing robust security measures for data protection
- Best practices for deploying and maintaining data warehouse solutions

Communication:

- Reporting on statistically complex issues

Competency Mapping

The tasks you will carry out in this week's challenge will contribute differently to the 11 competencies 10 Academy identified as essential for job preparedness in the field of Data Engineering, and Machine Learning engineering. The mapping below shows the change (lift) one can obtain through delivering the highest performance in these tasks.

Competency	**Potential contributions from this week**
Professionalism for a global-level job	Articulating business values
Collaboration and Communicating	Reporting to stakeholders
Software Development Frameworks	Using Github for CI/CD, writing modular codes, and packaging
Python Programming	Advanced use of python modules such as Pandas, Matplotlib, Numpy, Scikit-learn, Prophet and other relevant python packages
SQL programming	MySQL db create, read, and write
Data & Analytics Engineering	data filtering, data transformation, and data warehouse management
DBT	ELT & ETL for data transformation
Fast API	Create an Python API

Team

Tutors:

- Mahlet
- Elias
- Rediet
- Kerod
- Emitinan
- Rehmet

Key Dates

- Discussion on the case - 09:00 UTC on Wednesday 29 Jan 2025. Use #all-week7 to pre-ask questions.
- Interim Solution - 20:00 UTC on Friday 31 Jan 2025.
- Final Submission - 20:00 UTC on Tuesday 04 Feb 2025

Deliverables

Task 1 - Data scraping and collection pipeline

1. **Telegram Scraping:** Utilize the Telegram API or write custom scripts to extract data from public Telegram channels relevant to Ethiopian medical businesses. Use the following channels
    - https://t.me/DoctorsET
    - Chemed Telegram Channel
    - https://t.me/lobelia4cosmetics
    - https://t.me/yetenaweg
    - https://t.me/EAHCI
    - And many more from https://et.tgstat.com/medicine

2. **Image and Scraping:** Collect images from the following telegram channels for object detection:
    - Chemed Telegram Channel
    - https://t.me/lobelia4cosmetics

Steps:

- Use Python packages like:
    - For telegram: telethon
- Developing Telegram data extraction scripts or simply exporting content using the Telegram application.
- Storing Raw Data:
    - **Initial Storage**: Store the raw scraped data in a temporary storage location, such as a local database or files, before processing it further.
- Monitoring and Logging:
    - **Logging**: Implement logging to track the scraping process, capture errors, and monitor progress.

Task 2 - Data Cleaning and Transformation

- Data Cleaning:
    - Removing Duplicates
    - Handling Missing Values
    - Standardizing Formats
    - Data Validation
- Storing Cleaned Data
    - Database Storage

- DBT for Data Transformation:
  - Setting Up DBT: Install DBT (Data Build Tool) and set up a DBT project.
    ```
    pip install dbt
    dbt init my_project
    ```
  - Defining Models
    - Create DBT models for data transformation. DBT models are SQL files that define transformations on your data.
  - Run the DBT models to perform the transformations and load the data into your data warehouse.
    ```
    dbt run
    ```
  - Testing and Documentation: Use DBT's testing and documentation features to ensure data quality and provide context for the transformations.
    ```
    dbt test
    dbt docs generate
    dbt docs serve
    ```

- Monitoring and Logging:
  - **Logging**: Implement logging to track the scraping process, capture errors, and monitor progress.

Task 3 - Object Detection Using YOLO

- Setting Up the Environment:
  - Ensure you have the necessary dependencies installed, including YOLO and its required libraries (e.g., OpenCV, TensorFlow, or PyTorch depending on the YOLO implementation).
    ```
    pip install opencv-python
    pip install torch torchvision  # for PyTorch-based YOLO
    pip install tensorflow  # for TensorFlow-based YOLO
    ```

- Downloading the YOLO Model
    ```
    git clone https://github.com/ultralytics/yolov5.git
    cd yolov5
    pip install -r requirements.txt
    ```
- Preparing the Data
  - Collect images from the Chemed Telegram Channel, https://t.me/lobelia4cosmetics
- Use the pre-trained YOLO model to detect objects in the images.
- Processing the Detection Results
  - Extract relevant data from the detection results, such as bounding box coordinates, confidence scores, and class labels.

- Storing detection data to a database table.
- Monitoring and Logging:
    - **Logging**: Implement logging to track the scraping process, capture errors, and monitor progress.

Task 4 - Expose the collected data using Fast API

- Setting Up the Environment:
    - Install FastAPI and Uvicorn
      ```
      pip install fastapi uvicorn
      ```
- Create a FastAPI Application
    - Set up a basic project structure for your FastAPI application.
      ```
      my_project/
      ├── main.py
      ├── database.py
      ├── models.py
      ├── schemas.py
      └── crud.py
      ```
- Database Configuration
    - In the **database.py** configure the database connection using SQLAlchemy.
- Creating Data Models
    - In the **models.py** define SQLAlchemy models for the database tables.
- Creating Pydantic Schemas
    - In the **schemas.py** define Pydantic schemas for data validation and serialization.
- CRUD Operations
    - In the **crud.py** implement CRUD (Create, Read, Update, Delete) operations for the database.
- Creating API Endpoints
    - In the **main.py** define the API endpoints using FastAPI.

Tutorials Schedule

Overview

In the following, the colour **purple** indicates morning sessions, and **blue** indicates afternoon sessions.

Wednesday:

- Introduction to the challenge (Mahlet)
- Telegram scraping and Local database postgres (Elias).

Thursday:

- Extract, Load and Transform (Data Build Tool (DBT) (Kerod)

Friday:

- Object detection using YOLO (Rediet)

Monday:

- API development using Fast API (Emitinan)

Interim Submission

- Interim report - Covering task-1 and task-2
- Link to your GitHub.

Feedback

You may not receive detailed comments on your interim submission but will receive a grade.

Final Submission

- A blog post entry (which you can submit for example to Medium publishing) or a pdf report.
- Link to your Github code, and make sure to include screenshots demonstrating anything else you have done.

Feedback

You will receive comments/feedback in addition to a grade.

References

Web Scraping

1. https://realpython.com/python-web-scraping-practical-introduction/
2. https://realpython.com/beautiful-soup-web-scraper-python/
3. https://www.geeksforgeeks.org/python-web-scraping-tutorial/
4. https://scrapy.org/
5. https://www.selenium.dev/
6. https://docs.telethon.dev/en/stable/

DBT

1. https://www.getdbt.com/
2. https://docs.getdbt.com/docs/introduction
3. https://www.youtube.com/watch?v=C6BNAfaeqXY
4. https://www.startdataengineering.com/post/dbt-data-build-tool-tutorial/

YOLO

1. https://github.com/ultralytics/yolov5
2. https://docs.ultralytics.com/yolov5/tutorials/pytorch_hub_model_loading/#detailed-example
3. https://www.exxactcorp.com/blog/Deep-Learning/YOLOv5-PyTorch-Tutorial
4. https://github.com/LongxingTan/tfyolo
5. https://www.zehntech.com/real-time-object-detection-using-yolov5-and-tensorflow/
6. https://learnopencv.com/custom-object-detection-training-using-yolov5/

Fast API

1. https://fastapi.tiangolo.com/
2. https://fastapi.tiangolo.com/tutorial/first-steps/
3. https://realpython.com/fastapi-python-web-apis/

4. Pydantic and Fast API
5. https://medium.com/codenx/fastapi-pydantic-d809e046007f