

10 Academy Cohort B - Weekly Challenge: Week 5

## Scalable Data Warehouse for LLM Finetuning: API Design for High Throughput Data Ingestion and RAG Retrieval

# Overview

Business Need

Roots Tech Solutions is committed to enhancing its Natural Language Processing (NLP) capabilities for multiple African languages, particularly for Swahili, Yoruba, and Amharic languages. As part of this initiative, the company aims to develop a comprehensive data corpus to support various NLP applications, such as semantic search, content generation, chatbot support, sentiment analysis, speech recognition, and more.

The ability to process and understand Swahili, Yoruba, and Amharic text/audio accurately is critical for Roots Tech Solutions to develop innovative and competitive products. The current lack of extensive, high-quality text/audio datasets for these languages is a significant bottleneck. By collecting a vast amount of text/audio dataset from diverse online sources, including news websites, blogs, and social media platforms, the company can overcome this challenge.

The collected data will undergo cleaning and processing to ensure its quality and relevance. It will be stored in a structured database, making it easily accessible for various NLP tasks. Additionally, APIs will be developed to facilitate seamless integration and querying of the dataset, enabling the company to leverage this resource effectively.

This project is expected to significantly enhance Roots Tech Solutions' NLP capabilities, enabling the development of sophisticated African languages processing tools and applications, thereby providing a competitive edge in the technology market.

# Data

* You will be collecting the data from various data sources.

# Expected Outcomes

## Skills:

* Data Collection and Web Scraping:
  + Proficiency in web scraping techniques using tools like BeautifulSoup, Scrapy, and Selenium.
  + Expertise in identifying and extracting data from various online sources, including news websites, blogs, and social media platforms.
* Programming and Development Skills:
  + Proficiency in Python and Javascript (React) programming languages
  + Proficiency in SQL
  + Experience in developing APIs using frameworks like Flask or Django for data access and integration.
  + Proficiency in designing database schemas and managing structured data storage.
* Data Processing and Cleaning
* SQL and Database Management
* Experience in deploying complex software package using Docker and docker-compose
* UI/UX Design

## Knowledge:

* Natural Language Processing (NLP) Knowledge:
* Use of vector databases
* Machine Learning and AI Knowledge

# Team

Instructors:

* Yabebal
* Natnael
* Emtinan
* Rehmet

# Key Dates

* **Discussion on the case** - 9:00 UTC time on **20 May 2024**. Use #all-week-3 to ask questions.
* **Interim Submission** - 8:00 PM UTC time on **Wednesday 22 May 2024**.
* **Final Submission** - 8:00 PM UTC time on **Saturday 25 May 2024**

# Leaderboard for the week

There are 100 points available for the week.

20 points - community growth and peer support.

13 points - technical public and group-based RC channels

* Total number of messages (5)
* Total number of Mentions (3)
* Total number of DM connections (5)

7 points - community activities

* Number of messages in non-technical channels (4)
* On-time presence in Gmeet sessions (3)

30 points - presentation and reporting.

15 points - interim submission. PDF

15 points for the final submission. Blog entry or PDF with 5-8 pages.

50 points - Technical content

20 points - Interim submission

1. Github link submission (20)

30 points - Final submission

* Github Link submission (25)

Badges

Each week, one user will be awarded one of the badges below for the best performance in the category below.

In addition to being the badge holder for that badge, each badge winner will get +20 points to the overall score.

**Visualization** - the quality of visualizations, understandability, skimmability, choice of visualization

**Quality of code** - reliability, maintainability, efficiency, commenting - in the future this will be [CICD](https://en.wikipedia.org/wiki/CI/CD)

**An innovative approach to analysis** -using latest algorithms, adding in research paper content and other innovative approaches

**Writing and presentation** - clarity of written outputs, clarity of slides, overall production value

**Most supportive in the community** - helping others, adding links, tutoring those struggling

The goal of this approach is to support and reward expertise in different parts of the Machine learning engineering toolbox.

# Group Work Policy

This week, you are expected to complete the project with your assigned group. In the table below, your name is assigned to one of the groups we formed.

Please check out [this document](https://docs.google.com/document/d/16RPl4dwxBBghphPQxE9GMpC_r3AEsSKrWCZQgoLwQOA/edit?usp=sharing) for a guideline on how to connect to your cloud machine.

| Group Name (number) | Group Members | | Language | Machine Hostname |
| --- | --- | --- | --- | --- |
| Name | Email |
| Group 1 | Abdelrhman Yasir | [abdalrhmanalsh6@gmail.com](mailto:abdalrhmanalsh6@gmail.com) |  | [g1.10academy.org](http://g1.10academy.org) |
| Fatai Abdusalam | [abdusalamfatai60@gmail.com](mailto:abdusalamfatai60@gmail.com) |
| Getachew Abebe Agegnehu | [gechachin@gmail.com](mailto:gechachin@gmail.com) |
| Mistir Nigusse | [mistirnigusse0@gmail.com](mailto:mistirnigusse0@gmail.com) |
| Yohanes Teshome Kebede | [johnteshe13@gmail.com](mailto:johnteshe13@gmail.com) |
| Abraham Sahile | [abresh.agit@gmail.com](mailto:abresh.agit@gmail.com) |
| Selam Yoseph | [selamk499@gmail.com](mailto:selamk499@gmail.com) |
| Group 2 | Jolly Omere | [jollyomere@gmail.com](mailto:jollyomere@gmail.com) |  | [g2.10academy.org](http://g2.10academy.org) |
| Abraham Teka | [aberhamyirsaw@gmail.com](mailto:aberhamyirsaw@gmail.com) |
| Grace Nyutu | [graceynyutu@gmail.com](mailto:graceynyutu@gmail.com) |
| Melaku Alehegn | [melakualehegn34@gmail.com](mailto:melakualehegn34@gmail.com) |
| Henock Dessalegn | [henockgebriel@gmail.com](mailto:henockgebriel@gmail.com) |
| Biniyam Teshome | [se.biniyam.teshome@gmail.com](mailto:se.biniyam.teshome@gmail.com) |
| Blen Sleshi | [billu55sleshi@gmail.com](mailto:billu55sleshi@gmail.com) |
| Group 3 | Koomi Toussaint Amoussouvi | [toussaint.amoussouvi@yahoo.com](mailto:toussaint.amoussouvi@yahoo.com) |  | [g3.10academy.org](http://g3.10academy.org) |
| Abubeker Shamil | [abubekershamil123@gmail.com](mailto:abubekershamil123@gmail.com) |
| Michael George | [mikimare21@gmail.com](mailto:mikimare21@gmail.com) |
| Nyamusi Moraa | [nyamusiontita@gmail.com](mailto:nyamusiontita@gmail.com) |
| Ermias Asmare | [it.ermias.asmare@gmail.com](mailto:it.ermias.asmare@gmail.com) |
| Dawit Mulie | [daveget2020@gmail.com](mailto:daveget2020@gmail.com) |
| Eyerusalem Admassu | [eyerusad12@gmail.com](mailto:eyerusad12@gmail.com) |
| Group 4 | Temesgen Gebreabzgi | [temesgengebreab33@gmail.com](mailto:temesgengebreab33@gmail.com) |  | [g4.10academy.org](http://g4.10academy.org) |
| Addisu Alemu | [addisualemuj@gmail.com](mailto:addisualemuj@gmail.com) |
| Hillary Kipkemoi | [hillary6k@gmail.com](mailto:hillary6k@gmail.com) |
| Mahbubah Abdulhakim | [mahbubah.abdulhakim12@gmail.com](mailto:mahbubah.abdulhakim12@gmail.com) |
| Wandera Martin | [martinwandera@gmail.com](mailto:martinwandera@gmail.com) |
| Gilbert Tcheugoue Djissitchedjibril | [djibriltcheugoue@gmail.com](mailto:djibriltcheugoue@gmail.com) |
| Dawit Wubale | [dawittegegnwork@gmail.com](mailto:dawittegegnwork@gmail.com) |
| Group 5 | Hanna Mengistu | [hannasisay05@gmail.com](mailto:hannasisay05@gmail.com) |  | [g5.10academy.org](http://g5.10academy.org) |
| Bethelhem Mebratu | [baerem123@gmail.com](mailto:baerem123@gmail.com) |
| Jabez Kassa | [jabezkassa2022@gmail.com](mailto:jabezkassa2022@gmail.com) |
| Samson Leul | [samsonleul@gmail.com](mailto:samsonleul@gmail.com) |
| Tewodros Cheru | [tewodroscheru9@gmail.com](mailto:tewodroscheru9@gmail.com) |
| Grace Ogundaini | [graceogundaini@gmail.com](mailto:graceogundaini@gmail.com) |
| Mihret Kochito Wolde | [mihret.kochito16@gmail.com](mailto:mihret.kochito16@gmail.com) |
| Group 6 | Daisy Cherono | [jepchumbadaisy96@gmail.com](mailto:jepchumbadaisy96@gmail.com) |  | [g6.10academy.org](http://g6.10academy.org) |
| Dereje Hinsermu Senbatu | [derejehinsermu2@gmail.com](mailto:derejehinsermu2@gmail.com) |
| Joseph Njuguna | [josepnjuguuna@gmail.com](mailto:josepnjuguuna@gmail.com) |
| Selamawit Tibebu | [selamt4709@gmail.com](mailto:selamt4709@gmail.com) |
| Sheila Murugi | [tmurugi49@gmail.com](mailto:tmurugi49@gmail.com) |
| Mikias Wondim Gebre Medhin | [mikiaswondim0962939288@gmail.com](mailto:mikiaswondim0962939288@gmail.com) |

# Late Submission Policy

Our goal is to prepare successful learners for a global level job. At work, deadlines are sometimes very strict - either you do it before the deadline or the company loses a substantial opportunity. Moreover, the late communication behaviour (submission in 10 Academy can be considered as progress communication to team leads), blinds team leads and CEOs and is very determinantal in hindering the success of the company.

We have set our late submission as follows

* Submissions are accepted only within the 12 hrs window - 17:00 UTC - 7:00 UTC of the submission deadline
* Frequently late submissions (exceeding 6 total late submissions) will disqualify a person from the list of trainees 10 Academy recommends to partner employers.
* Badges will be rewarded for the cumulative on-time appearances (gmeet calls, on-time assignment submissions, and other places where being on-time is important)

## 

## 

## Possible Work Plan

### Decisions:

* Select group lead and other essential personals to facilitate efficient team work
* Select the language you will focus your effort to collect data for. The options are:
  + Swahili
  + Yoruba
  + Amharic
* Define the tech-stacks for your pipeline

### Work Plan

* Within your group, plan how you divide the work to group members
* Using Figma or similar tool, design your tech-stack and how data flows in your pipeline

### Stage 1: Initial Data Collection and Storage

- **Objective**: Establish foundational capabilities for collecting, storing, and accessing text data.

- Tools: Scrapy, BeautifulSoup, Selenium, MongoDB/PostgreSQL.

- Implementation:

* Identify and list potential data sources such as news websites, blogs, and social media.
* Develop web scraping scripts using Scrapy, BeautifulSoup, and Selenium to collect text data.
* Design and implement a database schema in PostgreSQL/MongoDb for storing the collected text data.
* Store the raw scraped data in the database.

### Stage 2: Data processing

- **Objective**: Clean, preprocess, and annotate the collected text data to ensure high quality for NLP applications.

- Implementation:

* Develop Python scripts to clean and preprocess the text data (e.g., remove HTML tags, and special characters).
* Filter data where possible to minimize data contamination by english or other languages that are not your focus. You may also consider creating a separate data table/category for english rich texts.
* Store the cleaned and processed data back into the database.

### Stage 3: API Development for Data Access

- **Objective**: Develop APIs to provide easy access to the collected and processed text data.

- Implementation:

* Develop RESTful APIs using Flask or FastAPI to enable querying and retrieving data from the database.
* Implement endpoints for searching and filtering text data based on keywords and metadata.
* Ensure efficiency in handling large-scale data access requests.

### Stage 4: Automation & Stream Processing

- **Objective**:

1. Implement an automated pipeline to ensure continuous data collection, cleaning, processing, storage, and access.

- Implementation:

* Setup workflow automation using Apache Airflow
* Define workflow DAG
* Containerize the workflow with Docker
* Automate deployment and monitoring

1. Handle data streams using a streaming processing library

- Implementation:

* Use the Faust library to setup data producer and consumer
* Measure the speed with which your pipeline can consume data in async mode
  + - [Locust - A modern load testing framework](https://locust.io/)

## Task 1: Data Collection and Storage

* Identify data sources:
  + Research and list potential sources such as news websites, blogs, social media platforms, online forums, and digital libraries.
  + Prioritize sources based on the quality and volume of content available.
* Set Up Development Environment
  + Install necessary tools and libraries (e.g., Scrapy, BeautifulSoup, Selenium).
  + Set up version control with Git and create a repository on GitHub.
* By using Scrapy, BeautifulSoup, and Selenium set up web scraping scripts
* Design an SQL/NoSQL schema for the raw data.
* Store raw data.

## Task 2: Data processing

* Retrieve the raw text data stored in PostgreSQL/MongoDb
* Normalize Text
  + Remove punctuation, and normalize whitespace.
* Remove HTML Tags and Special Characters
* Store the cleaned data.

## Task 3: API and Frontend Development for Data Access

Develop APIs and a frontend interface to provide easy access to the collected and processed data.

* Design API Endpoints
  + Define the endpoints needed for data access (e.g., search, filter, retrieve specific data).
* Implement API Using Flask/FastAPI
  + Create the API application with Flask or FastAPI.
* Develop the Frontend Interface Using React
  + Create a React application to interact with the API and display the data.
  + Create components for displaying data
  + Implement functionalities like
    - Search
    - Filter
    - Retrieve specific data

## Task 4: Data Annotation

Annotate the subset of the data to create labeled datasets for various Natural Language Processing (NLP) tasks and Supervised LLM finetuning, such as named entity recognition, sentiment analysis, and text classification.

* Set up annotation tools
* Prepare data for annotation
  + Prepare the cleaned and preprocessed text data for annotation.
  + Format the data appropriately for the annotation tool.
* Perform annotations
  + Use [Prodigy](https://prodi.gy/) to annotate the text data.
* Validate annotations
  + Ensure the quality and consistency of annotations.
* Store Annotated Data
  + Store the annotated data for use in NLP model training.

## Task 5: Automation

* Set Up Apache Airflow for workflow automation
  + Install and configure Apache Airflow to manage and orchestrate the different stages of the pipeline.
* Define workflow DAG (Directed Acyclic Graph)
  + Create a DAG to define the sequence of tasks for data collection, cleaning, processing, and storage.
* Containerize the Workflow with Docker
  + Create Dockerfiles for each component of the pipeline (data collection, cleaning, API).
* Create a docker-compose.yml file for orchestration
  + Use Docker Compose to define and manage the multi-container application.
* Implement CI/CD Pipeline with GitHub Actions
  + Set up GitHub Actions to automate testing, building, and deployment of the Docker images.
* Automate Deployment and Monitoring
  + Set up monitoring tools like Prometheus and Grafana to track the performance and health of the pipeline.

## Task 6: Blog Reporting

Write a blog-like report that details the process followed, the challenges faced, and lessons learned from this week’s challenge.

**N.B for reporting**

Your report should start with the Introduction, the overall body of your report, and then a Conclusion.

# 

# 

# Tutorials Schedule

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

## Monday: Project Overview and Data Collection Techniques

* Introduction to Week Challenge (Yabebal)
* Web Scraping and Data Collection Techniques(Emtinan)

Key Performance Indicators:

* Understanding week’s challenge
* Understanding Data Collection Techniques
* Understanding web scraping
* Ability to reuse previous knowledge

## Tuesday: Data Cleaning and Annotation

* Data preprocessing and cleaning for Amharic, Swahili, and Yoruba text.(Rehmet)
* Data annotation.(Rehmet)

## Wednesday: API and Frontend Development

* Develop APIs for data access.(Rehmet)
* Frontend development using react.(Rehmet)

## Thursday: Automation and Orchestration

* Automating the entire pipeline(Emtinan)
* Containerizing the application using Docker.(Emtinan)

## Friday: Monitoring

* Introduction to monitoring tools (Prometheus and Grafana).(Emtinan)

# Submission

## Interim: Due Wednesday 22 May 20:00 UTC

1. A PDF report with an overview of methodologies and tools used for data collection, including:
   * Detailed comparison between web scraping tools used.
2. Github link submission that demonstrates:
3. A working data collection script.
4. Database schema design.
5. Data loading to the database.
6. Work in progress for the front end.
7. Work in progress for backend API.

### 

### Feedback

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

## Final: Due Saturday 25 May 20:00 UTC

1. PDF document (to be published) or a published Blog link detailing the process followed. This should include:
2. The business objective of the project
3. The project design
4. The tech-stack used
5. The methodologies followed
6. The challenges faced
7. The results obtained
8. The lessons learned
9. Limitations and future plans
10. Github link submission that includes the following
11. Data Collection Scripts
12. Data Cleaning and Preprocessing Code
13. API and Frontend Code
14. Example annotations and how they are stored back into the database.
15. Dockerfile(s) for containerizing the different components of the project.
16. docker-compose.yml for orchestrating multi-container applications.
17. Bash/Python/Makefile scripts for automating the installation and setup process.
18. Continuous Integration and Continuous Deployment (CI/CD)
19. Logging

### Feedback

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

# References

Key References

* <https://www.selenium.dev/>
* <https://scrapy.org/>
* <https://beautiful-soup-4.readthedocs.io/en/latest/>
* <https://abe2g.github.io/am-preprocess.html>
* <https://data.mendeley.com/datasets/p74pfhz3yx/1>
* <https://prometheus.io/>
* <https://airflow.apache.org/>
* <https://prodi.gy/>
* <https://medium.com/@swalperen3008/what-is-dockerize-and-dockerize-your-project-a-step-by-step-guide-899c48a34df6>
* <https://grafana.com/docs/grafana/latest/setup-grafana/set-up-grafana-monitoring/>