**Capstone Project Concept Note and  Implementation Plan**

# **Project Title: Rainfall Based Drought prediction in Horn of Africa**

**Team Members**

1. Selam Kiflay

2. Hilina Amare

3. Abdi Gonfa

 4. Ketim Teklu

### Concept Note

#### 1. Project Overview

This project aims to develop a machine learning-driven model for forecasting droughts in the Horn of Africa using rainfall data, addressing the critical challenge of climate-induced agricultural disruptions. By predicting drought events through the Standardized Precipitation Index (SPI), our solution seeks to enable early warnings, supporting proactive measures in water management and food security. This aligns with Sustainable Development Goals (SDGs) 2 (Zero Hunger), 6 (Clean Water and Sanitation), and 13 (Climate Action), offering potential impacts in stabilizing livelihoods and enhancing resilience against escalating drought cycles in the region.

#### 2. Objectives

Our objectives are to:

* Develop an accurate forecasting model combining statistical and deep learning techniques to predict SPI-based drought events from rainfall data.
* Leverage freely accessible datasets, such as CHIRPS, to ensure scalability and applicability in data-scarce regions.
* Produce actionable visualizations, like drought risk maps, to communicate predictions effectively to stakeholders.
* Contribute to regional resilience by providing a framework for timely drought mitigation, advancing agricultural planning and resource allocation.

#### 3. Background

Droughts in the Horn of Africa, driven by erratic rainfall and exacerbated by climate change, threaten millions reliant on rain-fed agriculture. Existing solutions, such as seasonal weather forecasts or manual drought indices, often lack precision or scalability, relying on sparse ground data or coarse models. Machine learning offers a transformative approach by integrating diverse datasets (e.g., satellite rainfall) and capturing complex temporal patterns, surpassing traditional statistical methods in predictive power. Our project builds on this potential, tailoring a hybrid model to the region’s unique climatic and data constraints, aiming to bridge gaps in early warning systems.

#### 4. Methodology

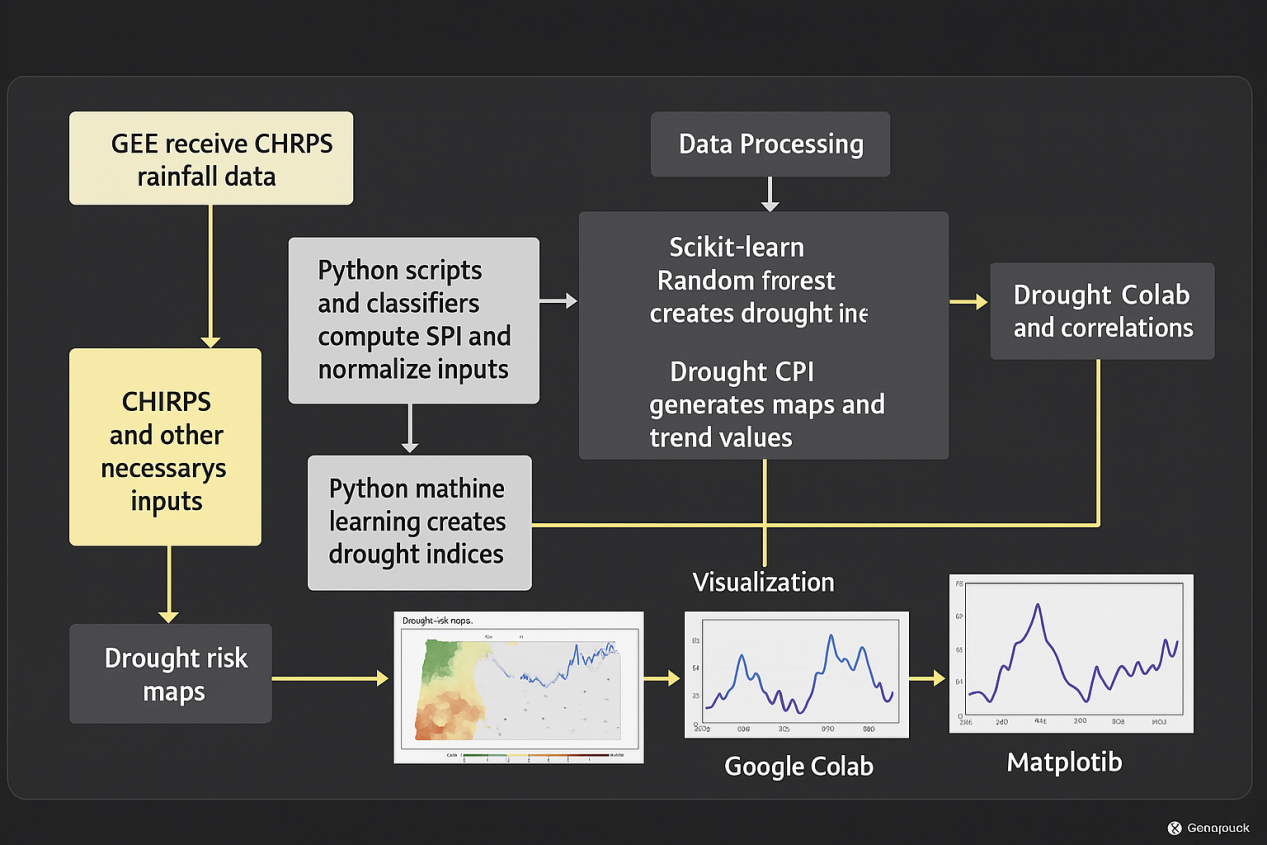
We will employ a hybrid machine learning approach, combining Scikit-learn’s Random Forest for baseline classification (drought vs. non-drought) with Keras-based Long Short-Term Memory (LSTM) networks to model temporal rainfall patterns for SPI prediction. Random Forest will process aggregated CHIRPS data to identify drought likelihood, while LSTMs will capture long-term dependencies in rainfall sequences, enhancing forecast accuracy. Google Colab will facilitate collaborative model development, and Google Earth Engine (GEE) will streamline data access. Matplotlib will generate visualizations, and SPI will be computed using Python-based statistical methods. Model evaluation will involve metrics like Mean Absolute Error (MAE) for SPI forecasts and accuracy for classification, with iterative tuning to optimize performance.

#### 5. Architecture Design Diagram

The system architecture comprises interconnected components:

* **Data Ingestion**: GEE retrieves CHIRPS rainfall data, preprocessed into time-series format.
* **Data Processing**: Python scripts compute SPI and normalize inputs for modeling.
* **Modeling**: Scikit-learn’s Random Forest classifies drought states; Keras’ LSTM predicts SPI values.
* **Visualization**: Matplotlib generates drought risk maps and trend plots.
* **Collaboration**: Google Colab integrates all processes, enabling team coordination.

**Diagram Description**:  
The diagram illustrates a pipeline where GEE feeds CHIRPS data into a preprocessing module, which outputs SPI and features to the Random Forest and LSTM models. Model outputs (classifications, SPI forecasts) flow to Matplotlib for visualization, all orchestrated within Colab’s cloud environment. Each component is designed for modularity, ensuring seamless interaction and scalability.



#### 6. Data Sources

We will utilize the Climate Hazards Group InfraRed Precipitation with Station data (CHIRPS), accessed via Google Earth Engine, providing high-resolution (0.05°) daily rainfall data from 1981 to present in NetCDF format, critical for SPI computation. The Standardized Precipitation Index (SPI) from FEWS NET, in CSV format, will serve as our drought indicator, capturing regional severity. These datasets, chosen for their open access and relevance to rainfall-driven drought patterns, require preprocessing steps like temporal aggregation and normalization to align with model inputs.

#### 7. Literature Review

Research highlights the efficacy of machine learning in drought forecasting. Poornima and Pushpalatha (2019) demonstrated LSTMs outperforming ARIMA in SPI prediction, leveraging rainfall data’s temporal dependencies. Nicholson (2014) used statistical methods for Horn rainfall prediction, noting limitations in sparse data settings. Xu et al. (2022) showed hybrid ARIMA-LSTM models balancing short- and long-term trends, inspiring our approach. Our project extends this work by integrating Random Forest and LSTM within a free, scalable framework, tailored to the Horn’s data constraints, enhancing regional predictive capacity.

### Implementation Plan

#### 1. Technology Stack

Our technology stack includes:

* **Programming Language**: Python, for its robust ML/DL libraries.
* **Libraries/Frameworks**: Scikit-learn (Random Forest), Keras (LSTM), Matplotlib (visualization), Pandas (data processing).
* **Platforms**: Google Colab (collaborative coding), Google Earth Engine (data access), pycharm and some other platforms.
* **Tools**: SPI computation via custom Python scripts, leveraging statistical methods.  
  This stack ensures accessibility, computational efficiency, and alignment with our forecasting objectives.

#### 2. Timeline

The project spans 12 weeks, with tasks distributed as follows:

* **Weeks 1-2: Data Collection and Preprocessing**
  + Task: Retrieve CHIRPS via GEE, compute SPI, normalize data.
  + Deliverable: Cleaned dataset in CSV format.
* **Weeks 3-5: Model Development**
  + Task: Implement Random Forest (Scikit-learn) and LSTM (Keras) in Colab.
  + Deliverable: Initial model prototypes.
* **Weeks 6-8: Training and Evaluation**
  + Task: Train models, evaluate using MAE and accuracy, tune hyperparameters.
  + Deliverable: Optimized models with performance reports.
* **Weeks 9-10: Visualization**
  + Task: Generate drought maps and trends with Matplotlib.
  + Deliverable: Visual outputs for stakeholder use.
* **Weeks 11-12: Deployment and Documentation**
  + Task: Finalize model integration in Colab, document findings.
  + Deliverable: Deployable model pipeline, final report.

**Task Distribution Matrix**:

| **Task** | **Member 1** | **Member 2** | **Member 3** | **Member 4** |
| --- | --- | --- | --- | --- |
| Data Collection | Lead | Support | - | - |
| Preprocessing | Support | Lead | - | - |
| Model Development | - | - | Lead | Support |
| Training/Evaluation | - | - | Support | Lead |
| Visualization | Lead | - | - | Support |
| Documentation | Support | Lead | Support | Support |

#### 3. Milestones

* **Milestone 1 (Week 2)**: Complete data collection and preprocessing, delivering a cleaned CHIRPS-SPI dataset.
* **Milestone 2 (Week 5)**: Develop initial Random Forest and LSTM models, achieving baseline performance.
* **Milestone 3 (Week 8)**: Optimize models, reaching MAE < 0.1 for SPI forecasts and accuracy > 80% for classification.
* **Milestone 4 (Week 10)**: Produce drought risk visualizations, validated against FEWS NET reports.
* **Milestone 5 (Week 12)**: Finalize deployable model pipeline and submit comprehensive report.
* Currently we are on the data collection and preprocessing , analyzing the datas, feasibility checking and other testing tasks for the complete project we are going to perform, since it will help us on the our way to develop the project in many ways and point us to the correct direction and more accurate predictions.

#### 4. Challenges and Mitigations

* **Challenge**: Data inconsistencies in CHIRPS (e.g., missing values).
  + **Mitigation**: we will try to apply interpolation techniques and cross-validate with FEWS NET SPI.
* **Challenge**: LSTM model complexity leading to overfitting or slow training.
  + **Mitigation**: Simplify architecture (e.g., fewer layers), use dropout, and leverage Colab’s cloud resources.
* **Challenge**: Limited internet affecting GEE access.
  + **Mitigation**: Pre-download CHIRPS data for offline preprocessing.

And also there may appear some challenges and problems on the way of working on the model and we will try to mitigate with the proper ways we could found.

#### 5. Ethical Considerations

Ensuring data privacy is paramount; CHIRPS and SPI are anonymized, posing no personal data risks. Bias in model predictions could arise from uneven data coverage across the Horn, potentially skewing forecasts for remote areas. We will mitigate this by validating outputs against diverse regional data and prioritizing transparency in model limitations. The project’s impact aims to benefit vulnerable communities, but we must ensure forecasts are accessible to local stakeholders, avoiding exclusion due to technical barriers.

#### 6. References

* Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Davis, A., ... & Ghemawat, S. (2016). TensorFlow: A system for large-scale machine learning. In 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16) (pp. 265–283).
* Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. Remote Sensing of Environment, 202, 18–27.
* Nicholson, S. E. (2014). The predictability of rainfall over the Greater Horn of Africa. Part I: Prediction of seasonal rainfall. Journal of Hydrometeorology, 15(3), 1341–1360.
* Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
* Poornima, S., & Pushpalatha, M. (2019). Drought prediction based on SPI and SPEI with varying timescales using LSTM recurrent neural network. Procedia Computer Science, 159, 2314–2323.
* Xu, D., Zhang, Q., Ding, Y., & Zhang, D. (2022). Application of a hybrid ARIMA-LSTM model based on the SPEI for drought forecasting. Environmental Science and Pollution Research, 29(3), 4128–4144.