Technical Brief: Knowledge Engineering Module

Project: Prediction of Extreme Climate Events

Knowledge Engineering Team

January 2, 2026

Abstract

This document outlines the theoretical foundations and technical implementation of the Knowledge Engineering module. It details how the system bridges the gap between raw Deep Learning predictions and actionable climate alerts using a Rule-Based Expert System.

Contents

1 Introduction

Our project aims to predict extreme climate events using historical data. While the **Deep Learning (DL)** module is responsible for forecasting future numerical values (e.g., Temperature, Humidity), it lacks semantic understanding. A neural network does not inherently know that $45°$C represents a dangerous heatwave; it only sees a number.

The **Knowledge Engineering (KE)** module serves as the "brain" of the system. It interprets the raw predictions from the DL module, applies meteorological rules, and infers whether an alert is necessary. This approach creates a **Hybrid AI** system that is both predictive and explainable.

2 Theoretical Foundations

To ensure all team members share a common vocabulary, we define the core concepts used in this module.

Ontology (The Vocabulary)

An ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts.
In our context: It acts as the schema for our knowledge. It defines what a "Heatwave" is, what properties it has (severity, duration), and how it relates to basic variables like "Temperature".

Inference Engine (The Logic)

The software component that applies logical rules to the knowledge base to deduce new information.
In our context: This is the algorithm that loops through the daily predictions. It checks conditions (e.g., `IF Temp > 40`) and triggers actions (e.g., `THEN Alert = True`).

Expert System

A computer system that emulates the decision-making ability of a human expert. It is composed of the Knowledge Base (Ontology + Rules) and the Inference Engine.

3 System Architecture: The Neuro-Symbolic Approach

We are implementing a **Neuro-Symbolic** architecture. This combines the statistical power of Neural Networks with the reliability of Symbolic Logic.

3.1 Workflow

1. **Input:** The LSTM (Deep Learning) model outputs a sequence of predicted variables for the next 7 days (e.g., T_{max}, $Humidity$, $WindSpeed$).

2. **Processing (KE Module):** The Inference Engine receives these numbers and maps them to our Ontology concepts.
3. **Reasoning:** The engine checks the data against pre-defined meteorological rules.
4. **Output:** The system generates a structured `ClimateEvent` object if a threat is detected.

4 Implementation Details

For our Minimum Viable Product (MVP), we are implementing the Ontology and Rules directly in Python.

4.1 The Ontology (Data Structure)

We define the structure of our "Knowledge" using Python Data Classes. This ensures strict typing and clear structure.

Listing 1: Python Class Definitions

```python
@dataclass
class MeteoFact:
    """Represents a single day of predicted data."""
    date: str
    tmax: float  # Maximum Temperature
    rhum: float  # Relative Humidity

@dataclass
class ClimateEvent:
    """Represents the inferred outcome (The Alert)."""
    event_type: str  # e.g., "Heatwave"
    severity: str    # "Moderate", "Critical"
    description: str # Explanation for the user
```

4.2 The Rules Engine (Logic)

We utilize **Forward Chaining** logic. We monitor specific thresholds to detect Heatwaves ("Canicules").

Rule 1: Critical Threshold (Daily)

Logic: If the maximum temperature exceeds the absolute critical limit.

$$\text{IF } T_{max} \geq 42°\text{C} \implies \text{Event} = \text{Critical Heatwave}$$

Rule 2: The "Dry Heat" Combo

Logic: Heat is more dangerous when combined with extremely low humidity (dehydration risk) or high humidity (wet bulb effect). For this MVP, we focus on dry heat.

$$\text{IF } (T_{max} \geq 38°\text{C}) \wedge (Humidity < 20\%) \implies \text{Event} = \text{High Risk}$$

Rule 3: Temporal Persistence (The 3-Day Rule)

Logic: A heatwave is officially defined by duration.

$$\text{IF } (T_{max} > 40°\text{C}) \text{ for } 3 \text{ consecutive days} \implies \text{Event} = \text{Persistent Heatwave}$$

5 Integration Strategy

To integrate this module into the wider application, the following data flow is required:

- **From Big Data/AI Team:** The ETL pipeline and LSTM model must provide clean, normalized data corresponding to the `MeteoFact` structure (specifically T_{max} and $Humidity$).
- **To Frontend Team:** The Web Interface should listen for `ClimateEvent` objects.
  - If `ClimateEvent` is **None**: Show standard green/blue dashboard.
  - If `ClimateEvent` exists: Display a "Red Alert" banner with the `description` field.

6 Conclusion

The Knowledge Engineering module provides the "safety net" for our AI. By formalizing meteorological rules into code, we ensure that our predictions result in accurate, explainable, and actionable warnings for the end user.