**Assignment: Adverse Drug Event Extraction Using Generative NLP Models (CADEC Dataset)**

**Objective**

Develop a **LangChain-based agentic NLP system** that extracts **adverse drug events (ADEs)** from **CADEC patient forum texts** using **only generative models from Hugging Face**. The system should iteratively verify extracted entities, retry if validation fails, and standardize the extracted drug names and adverse effects.

**Dataset: CADEC (A Corpus of Adverse Drug Event Annotations)**

The **CADEC dataset** consists of patient-written medical forum posts annotated with:

* **Drugs** (medications mentioned in user discussions)
* **Adverse Drug Events (ADEs)** (side effects caused by drugs)
* **Symptoms/Diseases** (related medical conditions)

🔹 **Access the dataset:** Available at [CADEC dataset repository](https://data.csiro.au/collection/csiro:10948?q=CADEC).

**Tasks**

**1. Data Preprocessing**

* Load **CADEC forum posts** and extract relevant sections.
* Apply **abbreviation expansion** using a **generative model** from Hugging Face.
* Tokenize text and normalize drug names.

**2. Medical Entity Extraction**

* Use a **Hugging Face generative model** to extract:
  + **Drugs (Medication Names)**
  + **Adverse Drug Events (ADEs)**
  + **Symptoms/Diseases**
* Return structured JSON output.

**3. Entity Standardization with UMLS**

* Map extracted entities to standardized medical terminologies using the **Unified Medical Language System (UMLS)** APIs:
  + **Drugs → RxNorm** (to unify different brand/generic names)
  + **ADEs & Symptoms → SNOMED CT** (to standardize adverse effects and symptoms)
* Steps for standardization:
  + - **Query UMLS API** with the extracted entity.
    - **Retrieve the best matching concept** (e.g., CUI - Concept Unique Identifier).
    - **Replace the extracted entity** with the standardized term from UMLS.

**4. Verification System**

Implement **three verification checks**:

1. **Format Verification** → Ensure valid JSON schema.
2. **Completeness Check** → Compare extracted entities with CADEC ground truth annotations.
3. **Semantic Similarity Check** → Use **cosine similarity** (via Sentence Transformers) to validate entity correctness.

**5. Agentic Iterative Correction**

* If verification fails, **call the generative model again with feedback**.
* **Limit retries to a maximum of 3 attempts**.
* Log **failures and improvements at each step**.

**Implementation Requirements**

* **Python** with LangChain and Hugging Face's transformers library.
* Use a **Hugging Face generative model** for:
  + Entity extraction
  + Abbreviation expansion
  + Standardization
* Extracted entities should be stored in **JSON format**.
* **UMLS API integration** for entity standardization.

**Evaluation Criteria**

| **Criteria** | **Description** |
| --- | --- |
| **Entity Extraction Accuracy** | Correctly identifies drugs, ADEs, and symptoms |
| **Entity Standardization (UMLS Mapping)** | Matches RxNorm, SNOMED CT |
| **Verification System** | Detects and corrects extraction errors |
| **Agentic Retries & Logs** | Retries extraction with feedback |
| **Code Quality & Documentation** | Readability and modularity |