Creation of Intensive Care Unit (ICU) Dictionary and LLM Prototype

Overview

Hi, In this repo you'll find:

- `README.md` → The explanation of the specific approach to build an ICU identifier dictionary and a **General Methodology** to ensure robust identifier building.
- `icu_data_dict_granular.json` → , A structured dictionary for Intensive Care Unit (ICU) data, currently under construction. It already includes a few key identifiers built following the outlined methodology.
- `Identifier_Builder.ipynb` → Notebook documenting my reasoning behind each identifier.
- `generate_random_documents.ipynb` → Generates synthetic documents to apply LLM classification for testing.
- `DLP_Product_Prototype.ipynb` → Implements an **LLM-based classification system** to identify sensitive data in documents, apply watermarking as a psychological deterrent, and label document metadata for enhanced security tracking.

Specific Method to ICU Identifiers

1️⃣ Consulting an ICU Expert for Real-World Validation

- Recognizing that source documentation alone cannot always be trusted, I took the **additional step of consulting an experienced ICU surgeon** to identify the most **high-risk** and **relevant** patient data points in Brazil.
- This ensured that the dataset had the **highest level of completeness and correctness** before implementing detection logic. Here are the most relevant datapoints to be considered:

DLP Classification

Category	Data Points (Brazilian_ Portuguese)	DLP Sensitivity Level	Potential Risks
Patient Identifiers (PHI)	Nome, Nº Atendimento, Nº Prontuário, Data Nascimento	🔴 High (PII/PHI)	Identity theft, unauthorized access
Hospitalization Data	DIH (Data da Internação Hospitalar), DUTI (Data da Admissão na UTI), LEITO	🟠 Medium	Exposure of admission dates could violate privacy
Insurance & Financial Data	Convênio, Seguro de saúde	🔴 High	Insurance fraud, financial abuse
Clinical & Medical Data	Diagnósticos, História Clínica, Ex Físico, Impressão Evolutiva	🔴 High (Sensitive PHI)	Misuse of medical history, blackmail risks
Medication & Treatment	Prescrição, Antibióticos, Drogas em BIC	🔴 High (PHI)	Prescription data leaks (e.g., opioids, controlled substances)
Vital Signs & Monitoring	Sinais Vitais, Parâmetros Ventilatórios	🟠 Medium	Exposure could lead to manipulation of health status data
Medical Devices & Procedures	Dispositivos Invasivos	🟠 Medium	Security risk if combined with patient identifiers
Medical Actions & Notes	Condutas	🟠 Medium	Internal procedures could reveal treatment plans
Laboratory Data	Labs	🔴 High (PHI)	Lab results often contain highly sensitive health markers
Dietary Information	Dieta	🟡 Low	Less critical but still part of patient records

2️⃣ Cross-Checking with Standardized Medical Taxonomies

To **enhance accuracy and interoperability**, I systematically started to review the ICU dictionary against international standards. This approach ensures **scalability** in case the dictionary needs to be adapted for **multilingual or cross-border use**.

- **LOINC** (*Logical Observation Identifiers Names and Codes*) → Standardized system for lab results & observations.
- **ICD** (*International Classification of Diseases*) → Universal classification of medical diagnoses.
- **CPT Codes** (*Current Procedural Terminology*) → Standardized list of medical procedures & ICU treatments.
- **SNOMED-CT** (*Systematized Nomenclature of Medicine - Clinical Terms*) → Comprehensive terminology for diagnostics & procedures.
- Additionally, I consulted **Pubmed** and other sources to understand the most common ICU cases.

3️⃣ Dual-Level Detection: Field-Based & Value-Based

To improve **structured and unstructured detection**, the system operates at two levels:

- **Field Identifier** → Detects whether a **certain tag/field** is present in the dataset.
- **Value Identifier** → Identifies **possible values** that the field can assume.

This ensures compatibility with both **structured formats** (e.g., JSON, XML, relational databases) and **unstructured text** (e.g., free-text clinical notes).

- The **Value Identifier** can be progressively expanded by integrating **taxonomy-based** detection.

Prototype for Data-at-Rest Protection

To **apply** the ICU dictionary in a real-world setting, I developed a prototype that performs **data-at-rest scanning**.

1️⃣ Traversing the Database with DFS

- The system recursively scans **all folders and files** (including subdirectories) using **Depth-First Search (DFS)** to efficiently locate **documents of interest**.
- It processes **PDF, DOCX, and CSV** files, extracting their contents for analysis.

2️⃣ Detecting & Marking Sensitive Data

If **sensitive data** is identified, the system **immediately applies two protective measures**:

1. **Watermarking** → Embeds a **"Sensitive" watermark** in documents, acting as a **psychological deterrent** to prevent accidental leaks or mishandling.
2. **Metadata Tagging** → Adds a "Sensitive" flag to document metadata for **enhanced tracking** (e.g., if the file is **copied, altered, uploaded, downloaded, or shared**).

General Methodology

This sections covers the steps that should be taken to guarantee the effective engineering of identifiers.

Linguistics-First Approach

Before implementing detection logic, the identifier must be **linguistically robust**. The methodology ensures that it accounts for **morphological structures, domain-specific terminology, and variations**.

1️⃣ Morphological Analysis

- **Word Stem** → Identifies the **fixed root** of words, utilizing **etymological understanding** and **NLP lemmatization tools**.
- **Prefixes, Postfixes & Inflections** → Examines how words **change endings** in different contexts (e.g., **declension, conjugation, gender, case variations**).
- **Domain-Specific Usage** → Recognizes **ICU-specific jargon, abbreviations, and medical terminology**.
- **Synonyms & Equivalent Concepts** → Accounts for **alternative terms** that may express **the same underlying meaning**.

-

2️⃣ Translation into Symbolic Language (Regex, NLP, Encoding)

Once the linguistic patterns are understood, the next step is converting them into **structured identifiers**.

Handling Variations & Anomalies

- **Typo & Variation Handling** → Implements **Regex-based typo correction** and **fuzzy matching algorithms** (Levenshtein distance, cosine similarity, etc.).
- **Encoding Challenges** → Accounts for **special character removal** (e.g., "ação" → "acao") to ensure **ASCII compatibility**.
- **Whitespace Sensitivity** → Ensures detection is **robust against spacing inconsistencies** (e.g., "NomePaciente" vs. "Nome Paciente").

Balancing Generalization & Precision

- The key principle: **"As general as possible, as specific as necessary."**
- This prevents **overfitting** to ICU-specific language while maintaining **high precision**.

3️⃣ Iterative Refinement

- **Regex Prototyping & Debugging** → Live-testing with **Regex101** to fine-tune pattern structure.
- **LLM-Augmented Review** → Although **LLMs struggle with low-level character recognition**, they are valuable for **spotting inconsistencies** and **suggesting missing cases**.

Stress-Testing the Identifier

The goal is to **expose the identifier's limitations and strengths** through a **multi-faceted stress-testing approach**.

1️⃣ Positive Stress-Test (PST)

- **Goal:** Ensure the identifier is **general enough**.
- **Method:** Generate a **large, randomized dataset** with **valid names, terms, and patterns**. Test against the identifier and verify **successful matches**.

2️⃣ **Negative Stress-Test (NST)**

- **Goal:** Ensure the identifier is **specific enough**.
- **Method:** Generate a **dataset of false positives**—texts that **should NOT match** the pattern. Run tests to **eliminate false detections**.

3️⃣ **Moriarty Stress-Test (MST)**

- **Goal:** Simulate **adversarial evasion attempts**.
- **Method:** Take a step back and ask:

> "If I wanted to bypass this detection, how would I do it?"

Actively **attempt to break the identifier**. This can involve:

✅ **Creative obfuscation techniques**

✅ **Pattern manipulation (spacing, special characters, typos)**

✅ **Unstructured or fragmented text formatting**

The best improvements **often go beyond Regex alone**—leading to **enhanced hybrid detection strategies**.

4️⃣ **Time Stress-Test (TST)**

- **Goal:** Ensure the identifier remains valid over time.
- **Method:**
- Tag each identifier with a **time sensitivity level** (e.g., *"Stable", "Likely to Change", "Requires Review"*).
- Implement a **scheduled review process** to reassess identifiers flagged as **unstable or evolving**.
- Use **automated monitoring** to detect changes in medical taxonomies (ICD, LOINC, etc.).

Regex Usage & Implementation

Regex is a core component of structured pattern detection, but its application requires **deliberate selection of functions and syntax** depending on the context.

- 1️⃣**Understand the available methods**:
  - `findall()` → Extracts **all** matches in a given text.
  - `search()` → Finds the **first** match, useful for presence checks.
  - `match()` → Checks **only** at the start of the string.
  - `fullmatch()` → Ensures the **entire string** matches the pattern.
  - `compile()` → Pre-compiles the regex for **optimized reuse**.
- 2️⃣**Cross-check regex behavior across different languages**:
  - Python, Go, and JavaScript each have slight **variations in regex handling**.
  - Some engines allow **lookbehind assertions**, while others don't.
  - Always test for **edge cases** where regex might fail due to compilation quirks.

Preventive Alignment with Client's DLP Readiness

Depending on how well **the organization enforces DLP policies**, there may be **pre-existing structures** that can **enhance detection accuracy**.

✅ **Fixed Identifier Standards** → If certain **IDs** (e.g., patient IDs, case numbers) follow a strict **numeric or alphanumeric structure**, regex rules can be optimized to enforce **expected formats only**.

✅ **Standard Document Titles & Tags** → If policies **require sensitive documents** to be **explicitly labeled**, DLP detection can **leverage these fields** instead of relying solely on content analysis.

✅ **Proactive Policy Integration** → Work with the **client's compliance team** to **align detection with existing data governance policies** (e.g., ensuring system-generated reports include structured metadata).

By aligning **preventive strategies** with **automated detection**, we can **reduce false positives** while **reinforcing security** at the source.

Next Steps & Future Considerations

- **Expand Taxonomy Matching** → Integrate with **ICD-10, LOINC, and SNOMED-CT databases** for automated cross-referencing.

- **Enhance Contextual Classification** → Fine-tune **LLM-assisted detection** to classify sensitive data **beyond simple keyword matching**.
- **Optimize Computational Performance** → Reduce **false positives & negatives** while maintaining **high efficiency** for large-scale data scanning.

Final Thoughts

This methodology **ensures a rigorous, linguistically-grounded approach** to ICU data classification. By **combining domain expertise, structured taxonomies, and hybrid detection techniques**, the project lays a **strong foundation for robust DLP solutions** in healthcare.

Excited to refine this further based on feedback!