A LLMOps-Driven Framework for Clinical Data Harmonization

Alberto Marfoglia[1,*], Antonio Robustelli[1], Christian D'Errico[1], Sabato Mellone[2] and Antonella Carbonaro[1]

[1]*Department of Computer Science and Engineering, University of Bologna, Italy*
[2]*Department of Electrical, Electronic and Information Engineering "Guglielmo Marconi", University of Bologna, Italy*

Abstract

The rapid growth of clinical data, driven by advances in medical research and digital health technologies, presents major challenges in ensuring data interoperability, standardization, and management. Standards such as Fast Healthcare Interoperability Resources (FHIR) are essential for enabling seamless data exchange. However, converting raw data into standardized formats remains a complex and resource-intensive task. Existing solutions often rely on manual processes or rigid rule-based systems, which are time-consuming, error-prone, and difficult to scale. To address these limitations, we propose a modular framework that employs Natural Language Processing (NLP) to streamline the FHIR mapping. Additionally, it integrates Large Language Model Operations (LLMOps) principles to automate and monitor the lifecycle of models involved in the data transformation. The framework consists of three core modules: (1) extraction of relevant clinical variables from heterogeneous data sources, (2) validation for anomaly detection and compliance with healthcare standards, and (3) assisted mapping of variables to FHIR resources. We evaluate the framework by applying it to an existing clinical data harmonization pipeline. Compared to the baseline process, our approach achieves a 59% reduction in time. This result underscores the potential of NLP-assisted frameworks to improve scalability, reliability, and efficiency in clinical data standardization.

Keywords

Clinical Data Harmonization, FHIR (Fast Healthcare Interoperability Resources), Digital Health, Natural Language Processing (NLP), Large Language Model Operations (LLMOps)

1. Introduction

The healthcare landscape is rapidly evolving, driven by advancements in medical research and digital technologies, leading to an unprecedented surge in clinical data. Effectively managing and sharing this data across heterogeneous healthcare organizations remains a critical challenge impacting patient care, medical research, and policy development [1]. Seamless data exchange requires both structural and semantic interoperability, which involves the adoption of standardized vocabularies, formal data descriptions, and machine-readable formats. Consequently, tools leveraging contemporary clinical standards for data mapping have become essential, often enabling semi-automated conversion workflows that would otherwise require substantial manual effort [2].

Interoperability is the foundation of modern digital health initiatives [3, 4], facilitating the prospective curation of data, efficient communication, and the secondary use of real-world clinical data. The European Health Data Space (EHDS) exemplifies its economic and strategic value, aiming to establish a unified health data ecosystem projected to save €11 billion over ten years [5].

Among existing healthcare data standards, the Fast Healthcare Interoperability Resources (FHIR) framework, developed by Health Level Seven International (HL7), has emerged as a leading approach for structuring and exchanging clinical information. FHIR ensures flexible and modular data representation, facilitating interoperability across clinical trials, hospital information systems, and public

MLOps25: Workshop on Machine Learning Operations. October 25, 2025, Bologna, Italy

[*] Corresponding author.

alberto.marfoglia2@unibo.it (A. Marfoglia); antonio.robustelli2@unibo.it (A. Robustelli); christian.derrico2@unibo.it (C. D'Errico); sabato.mellone@unibo.it (S. Mellone); antonella.carbonaro@unibo.it (A. Carbonaro)
00009-0000-5857-2376 (A. Marfoglia); 0000-0003-3423-3374 (A. Robustelli); 0009-0002-5784-0492 (C. D'Errico); 0000-0001-7688-0188 (S. Mellone); 0000-0002-3890-4852 (A. Carbonaro)

health networks [6]. Its cost-effectiveness, improved data quality, and analytical flexibility make it particularly advantageous for reusing medical data in the real world [7]. Increasingly, FHIR is being adopted in specialized healthcare domains, from chronic disease management and patient-centered applications [8] to modular platforms that unify heterogeneous data sources and create standardized Digital Twins [9, 10].

Despite the growing adoption of FHIR, converting raw clinical data into standardized formats remains a complex, labor-intensive process. Traditional transformation approaches rely heavily on manual effort and static rules, which limits scalability and introduces inconsistency [11]. Natural Language Processing (NLP), particularly through Large Language Models (LLMs), offers a promising approach to automate core tasks such as clinical variable extraction, data validation, and resource mapping [12, 13].

However, LLM-based systems present substantial barriers when transitioning from research to real-world deployment. The healthcare domain represents a particular case, where data privacy, regulatory compliance, and model governance are non-negotiable [14]. Consequently, many promising prototypes remain confined to the experimental phases due to the lack of robust operational infrastructure and interdisciplinary collaboration [15]. In this context, recent engineering paradigms, such as Machine Learning Operations (MLOps), offer a practical foundation for ensuring the reliability of AI, thereby addressing these concerns [16, 17]. Based on MLOps features, such as automation, continuous monitoring, and reproducibility, Large Language Model Operations (LLMOps) frameworks represent a promising solution to satisfy the operational demands of the LLMs lifecycle, including prompt versioning, input-output management, and real-time risk mitigation of hallucinations and performance drift [18].

To address these technical and operational limitations, we present FLEX-LLM-CARE (FHIR Language Extraction and Transformation with LLMs for Clinical Automation and Reasoning Engine). This modular LLM-based transformation framework leverages LLMOps techniques to streamline the standardization of clinical data into FHIR resources. The proposed solution enhances three key harmonization steps: (1) the extraction of clinical variables from both structured and unstructured data; (2) the validation through anomaly detection and compliance with clinical guidelines; and (3) the transformation of validated content into interoperable FHIR resources.

To demonstrate the effectiveness of the proposed framework, we apply it to an existing standardization pipeline [2] to enhance the automation level of the overall process. Our results show that FLEX-LLM-CARE reduces the total manual effort required for data harmonization by 59%, equivalent to saving over 200 hours of experts' labor.

These findings highlight FLEX-LLM-CARE's potential to make large-scale FHIR adoption more practical and cost-effective. Beyond reducing manual effort, the integration of LLMOps introduces essential capabilities that support long-term sustainability and reliability. Looking ahead, tools like FLEX-LLM-CARE can help healthcare organizations adopt data standards more easily, leading to faster insights, improved data sharing, and more connected, interoperable health systems.

The remainder of this paper is structured as follows: Sec. 3 reviews related NLP-based standardization efforts. Sec. 2 introduces LLMOps and the baseline pipeline. Sec. 4 details our framework modules. Sec. 5 presents the experimental results and Sec. 6 concludes with a discussion of findings and future directions.

2. Background

This section provides an overview of the concepts underlying FLEX-LLM-CARE. We first summarize the main characteristics of LLMOps, which represent the set of concepts adopted to improve the automation of our proposal. Then, we briefly recall the high-level structure of the standardization pipeline on which the framework is applied.

2.1. Large Language Model Operations

Constructing LLMs is a complex task that involves vast amounts of data, High-Performance Computing (HPC) architectures, and targeted fine-tuning steps [19]. The datasets required to train LLMs typically contain so many parameters that manual data quality checks become impractical. Consequently, LLMs can suffer from biases, hallucinations, and outdated knowledge, resulting in inaccurate or misleading outcomes. For this reason, engineering paradigms based on MLOps are fundamental to face these issues by ensuring automation, continuous monitoring, and reproducibility [16, 17]. In the context of LLMs, the solution is represented by LLMOps, which, according to the definition provided by Diaz-De-Arcaya et al. [19], is an extension of MLOps specifically tailored to manage the LLMs lifecycle effectively.

Since the management of LLMs is more complex than ML models, Shan and Shan [18] defined a conceptual framework named 4D that, as shown in Fig. 1, takes its name from the following four steps:

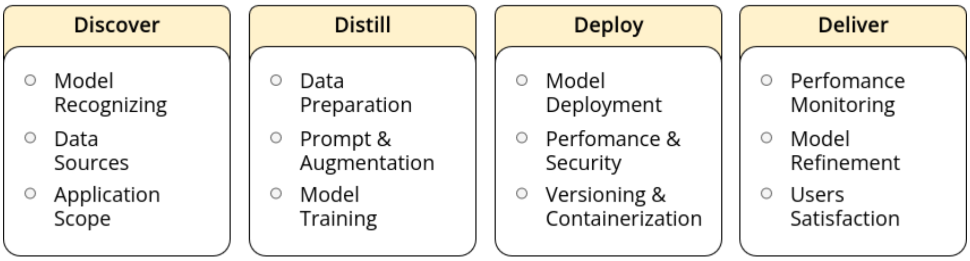

Figure 1: High-Level representation of the 4D framework.

- **Discover**: it recognizes the need for an LLM and explores its potential use cases. This involves examining recent developments in LLM technology, understanding their capabilities, and assessing how they can solve specific issues or improve existing workflows. Hence, this step identifies relevant data sources, sets objectives, and defines the application scope.
- **Distill**: it prepares and refines the data employed for the training process. Since data quality and variety are crucial for the model's performance, this step involves data cleaning, structuring, and augmentation. It also includes the initial model training, in which the system learns how to generate outputs or predictions based on the distilled data.
- **Deploy**: it integrates the LLM into the operational environment by making it a part of a broader system. Hence, this step involves establishing the technical infrastructure, addressing performance and security requirements, and ensuring smooth interaction with existing tools. It also focuses on version control, containerization, and API connectivity.
- **Deliver**: it delivers the LLM in production. This involves monitoring its performance in real-world scenarios and refining it continuously based on user feedback and new input data. This step also includes evaluating the LLM's impact on business results and user satisfaction by making ongoing adjustments to optimize its effectiveness.

Since the 4D framework offers a systematic approach for LLM lifecycle management, we adopt it to automate the data standardization process within our proposal. Note that, due to its conceptual nature, this framework comprises several substeps. However, to simplify the discussion, we refer to [16, 17] for more details and specifications.

2.2. The considered pipeline

We evaluated the effectiveness of our proposed framework and its modular components (as described in Sec. 4) by applying it to optimize the FHIR mapping workflow of a pipeline introduced in a previous study [2]. This workflow follows a classical Extract-Transform-Load (ETL) architecture and comprises the following five sequential modules:

- **Input**: gathers input data from a specific source, without applying restrictions on the data model and format. Additionally, this module addresses data security and regulatory compliance by ensuring the quality, integrity, and confidentiality of data.

- **Refinement**: preprocesses the gathered input data by ensuring a uniform output format (e.g., JSON). To this end, this module removes missing data and structures the information to enable seamless conversion operations in subsequent steps.
- **Mapping**: converts the refined input data to the target data model (e.g., FHIR). To this end, this module employs a templating strategy;
- **Validation**: it performs all the procedures to check the conformity of output resources with the standard target data model. In the event of undesirable outcomes, this module immediately performs roll-back actions to ensure the system's state remains intact.
- **Publishing**: stores the validated data in an accessible repository. This module also provides essential debugging functionalities like querying, exporting, and logging.

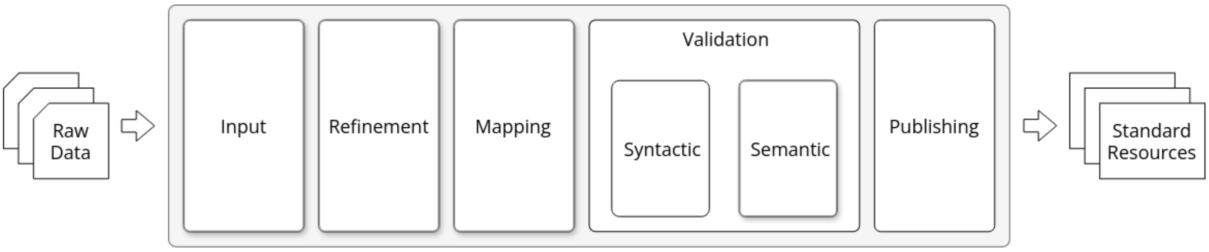

Figure 2: Modular representation of the considered pipeline, organized according to the ETL design [2].

Despite its modularity, the pipeline exhibits several limitations due to its reliance on manual intervention. Specifically, the *Input* module requires domain experts to manually define the relevant tables and annotate each field with appropriate semantic labels and ontologies. The *Refinement* module necessitates expert consultations to identify and resolve missing or inconsistent values. The *Mapping* module depends on curated templates developed by experts, while the *Validation* module operates using preconfigured static FHIR profiles.

To address these limitations and enhance scalability and consistency, we integrate FLEX-LLM-CARE into the pipeline. This integration aims to automate critical phases of the standardization process, thereby minimizing manual effort and reducing the risk of data inconsistencies. A comparative analysis of processing times between the original pipeline and the enhanced version incorporating our framework is presented in Sec. 5.2.

3. Related Work

Efforts to standardize clinical data have led to the development of several pipelines aimed at enhancing interoperability and transforming heterogeneous data formats into structured models such as HL7 FHIR [6, 7, 8]. For example, Bennett et al. [20] converted the MIMIC-IV dataset into FHIR, creating a widely used resource for research. Montazeri et al. [21] developed a FHIR-based dataset to support cardiovascular CPOE integration within Electronic Health Record (EHR) systems.

Beyond these datasets, several approaches have targeted the technical aspects of FHIR mapping. Sinaci et al. [22] proposed a GUI-based method, Simon et al. [23] developed a metadata repository enabling automated data conversion through a REST API, and Marfoglia et al. [2] introduced a modular ETL pipeline using template-based mappings to enhance platform independence and reusability.

While these approaches mark progress toward interoperability, key limitations persist. Most notably, the lack of full automation necessitates manual data curation and validation, which slows down the mapping process and increases the risk of errors. Additionally, reliance on custom scripts and opaque logic reduces portability across healthcare contexts.

Natural Language Processing (NLP) and, more recently, Large Language Models (LLMs), have shown promise in healthcare for processing unstructured data [24, 12, 13]. For instance, NLP has supported clinical trial automation by extracting patient data from EHRs and identifying adverse drug events [25].

Within Clinical Decision Support Systems (CDSS), Klug et al. [26] applied NLP to extract actionable insights from clinical notes. Remmer et al. [27] leveraged NLP for the multi-label classification of medical summaries, combining clinical and linguistic embeddings. Instead, Gulum et al. [28] combined Deep Learning (DL) with NLP to enhance cancer decision-making, providing more accurate diagnoses and personalized treatment recommendations. Despite their success, these NLP solutions primarily address classification, summarization, or prediction tasks rather than the challenge of converting structured clinical data into FHIR-compliant formats.

Moreover, deploying AI systems in real-world healthcare settings requires strict attention to governance, reproducibility, and regulatory compliance [14]. In response, MLOps has emerged to support continuous monitoring and operational scalability. LLMOps, an evolution of MLOps for LLMs, extends these capabilities to the management of large-scale language models [16, 17]. In this context, we propose FLEX-LLM-CARE, a modular LLM-based transformation framework that applies LLMOps principles to streamline the FHIR data standardization process. In contrast to prior work (see Table 1), our approach minimizes manual intervention, avoids static configurations, and enables more scalable, reusable, and transparent data standardization pipelines.

Ref.	Automation	Validation	Governance	Scalability	Limitation
[20]	Manual configuration with internal mapping logic	None	None	Low	Hard-coded mappings; non-reusable across datasets
[22]	GUI-based, expert-driven mapping	Expert-mediated via constrained FHIR profiles	None	Low	Dependent on expert availability; rigid profile constraints
[2]	Semi-automated via predefined templates	Static validation using fixed profiles	None	Medium	Portable templates but lack adaptability to new domains
Ours	LLM-assisted dynamic mapping with NLP-based extraction	Built-in anomaly detection & FHIR compliance checks	LLMOps lifecycle	High	Higher complexity and resource demand

Table 1
Comparative summary of FHIR mapping frameworks across key dimensions.

4. The proposed framework

This section introduces the proposed FLEX-LLM-CARE framework, which leverages recent advances in LLMs to enable the automated standardization and semantic enrichment of clinical data in compliance with the FHIR specification. To this end, we define three core modules, which are respectively focused on: (1) the extraction of clinical variables from both structured and unstructured data; (2) the validation through anomaly detection and compliance with clinical guidelines; and (3) the mapping of validated content into interoperable FHIR resources. Subsequently, we describe how LLMOps (i.e., the 4D framework) is applied in FLEX-LLM-CARE to automate the data mapping. Fig. 3 illustrates the FLEX-LLM-CARE framework, highlighting its modules, employed strategies, and considered input and output data formats.

4.1. Clinical Data Extraction Module (CDE-M)

CDE-M extracts clinical concepts from unstructured and semi-structured text using domain-specific Named Entity Recognition (NER) models. Then, a specialized LLM links the extracted entities to standardized clinical ontologies. In detail, this association is possible by combining the LLM with the Retrieval-Augmented Generation (RAG) technique. RAG integrates external facts by retrieving relevant information from a pre-built knowledge base to improve the accuracy of the output. In

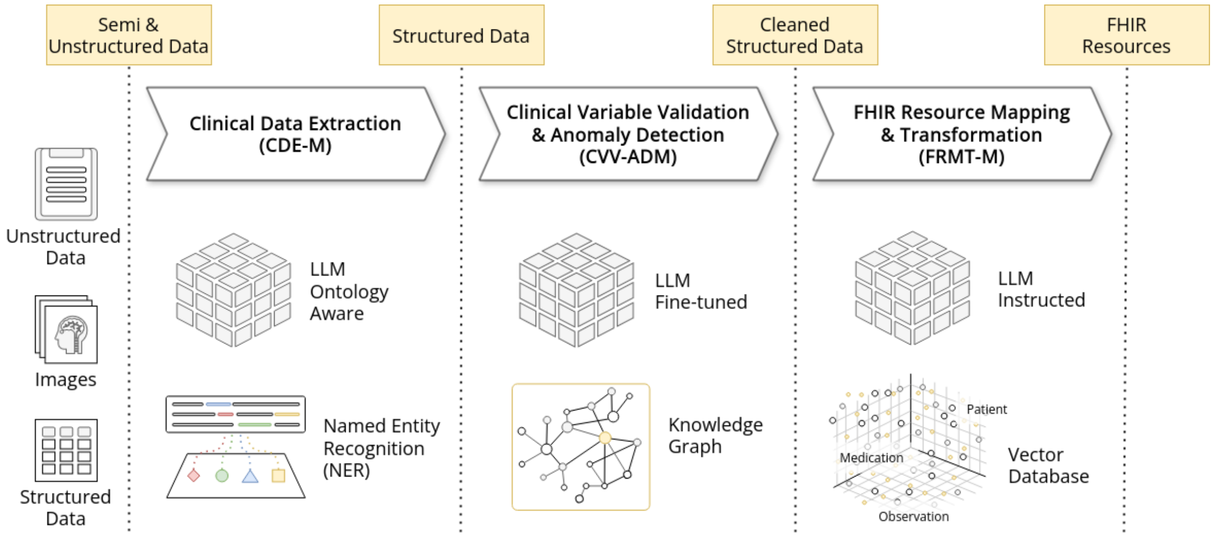

Figure 3: Overview of the FLEX-LLM-CARE and its three modules: CDE-M, CVV-ADM, and FRMT-M.

this case, the knowledge corresponds to the embedding representation of clinical ontology concepts stored in a vectorial database. Therefore, this approach resolves ambiguities and improves contextual understanding, making the extracted data more usable for seamless conversion into FHIR resources.

4.2. Clinical Variable Validation and Anomaly Detection Module (CVV-ADM)

CVV-ADM ensures that the structured data is clinically consistent and semantically valid within the related medical context. In detail, its core is represented by an LLM capable of understanding clinical logic, domain-specific terminology, and guideline-based reasoning. For this reason, CVV-ADM supports two sequential validation strategies:

- **Guideline-Aware**: the employed LLM directly evaluates the structured data by referencing medical best practices and clinical guidelines. Moreover, this strategy validates data by checking anomalies, such as incompatible medications, implausible lab values, missing context, and elements falling outside the expected clinical norms;
- **Ontology-Driven**: it converts validated data into graph-based structures (i.e., semantic nodes and relationships) linked to medical ontologies, enriching their semantics. This strategy enables symbolic reasoning and logical inference across the graph, allowing the detection of inconsistencies, redundancies, or missing connections.

4.3. FHIR Resource Mapping and Transformation Module (FRMT-M)

FRMT-M streamlines the mapping process of validated and cleaned data to the corresponding FHIR resources. In detail, this module employs an LLM that can learn the mapping logic through an In-Context Learning (ICL) approach or a targeted fine-tuning. Moreover, the LLM integrates a RAG mechanism that accesses a vector database populated with the FHIR documentation, related specifications, and examples of validated mappings. Therefore, this approach enables the generation of outputs validated against authoritative sources, thereby enhancing the explainability and reliability of the resulting FHIR mapping.

4.4. LLMOps trought the 4D framework

Since the defined modules introduce new challenges regarding intermediate datasets, additional data structures, and various LLMs, LLMOps becomes paramount to improve the automation level of the entire FHIR mapping process. To this end, according to the definition provided in Sec. 2.1, we describe how the 4D framework enhances the defined modules (i.e., CDE-M, CVV-ADM, and FRMT-M):

- **Discover**: Since we face different specific tasks (i.e., data extraction from unstructured text, data validation to detect anomalies, and FHIR mapping), it becomes crucial to select the most appropriate LLM. To this end, referring to existing LLM benchmarkings, such as the rankings provided by Hugging Face, can help choose the most suitable model for our tasks.
- **Distill**: Since it is essential to improve the prediction logic and the possible training of the LLMs chosen, selecting the data cleaning, structuring, and augmentation techniques becomes pivotal. For example, for the CDE-M and FRMT-M, we can use the RAG and prompt engineering techniques previously mentioned. In the case of CVV-ADM, since the LLM model is fine-tuned on medical knowledge and anomaly detection tasks, we can employ PEFT, LoRA, or QLoRA to adapt models with minimal computational resources efficiently. When training from scratch or at scale, Composer (by MosaicML) and datasets from LLMDataHub can be used.
- **Deploy**: Since we use different LLMs and techniques, the resulting hyperparameter combinations and model versions can be numerous. For this reason, we ensure that all experiments are rigorously tracked throughout the deployment process. It is essential to leverage open-source platforms such as MLFlow for lifecycle management, and orchestrators like Kubeflow or Apache Airflow for pipeline automation. Containerization with Docker and orchestration via Kubernetes further enhances scalability and portability across deployment environments.
- **Deliver**: Since high-quality clinical data is often limited and not directly accessible, continuous post-deployment monitoring is essential to assess model robustness and reliability over time. To this end, we can integrate monitoring tools such as Evidently AI for model evaluation and drift detection. Prometheus and Grafana can be used for real-time metric collection and visualization. Furthermore, LangSmith offers tools for debugging and testing LLM-based applications, while OpenLLMetry and LangKit enhance observability and governance by extracting key metrics from LLM input/output behavior.

However, due to the numerous application scenarios, the mentioned technologies represent only a small subset of the existing ones. For this reason, we refer to [18, 29] for a more exhaustive overview.

5. Results

This section presents the evaluation of FLEX-LLM-CARE in automating clinical data harmonization using the FHIR standard. We begin by detailing the technical implementation of each module—CDE-M, CVV-ADM, and FRMT-M—within the data standardization pipeline shown in Fig.2. We then assess the practical impact of our framework by comparing it to the baseline pipeline introduced by Marfoglia et al.[2], focusing on the harmonization of a public dataset [30].

5.1. Modules implementations

According to the limitations discussed in Sec. 2.2, we applied FLEX-LLM-CARE's modules to the standardization pipeline depicted in Fig. 2. To this end, as shown in Fig. 4, we made the following associations: (CDE-M $\mapsto$ Input), (CVV-ADM $\mapsto$ Refinement), and (FRMT-M $\mapsto$ Mapping).

Fig. 4 allows us to appreciate how the application of FLEX-LLM-CARE enhances the automation level of the Input, Refinement, and Mapping steps. This is followed by the Validation and Publishing steps, which, although fully automated, depend on the results of the first three steps. Finally, the Semantic module is highlighted in orange but is not explored in this study, as it is outside the scope.

In detail, to extract clinical concepts from the unstructured and semi-structured text, we implemented the CDE-Module by employing BioBERT [31] to identify clinical entities. Then, we mapped such entities into standardized ontologies using BioMistral-7B [32], a biomedical LLM capable of interpreting column headers and field contents. To support this mapping, we also integrated a RAG technique to perform a

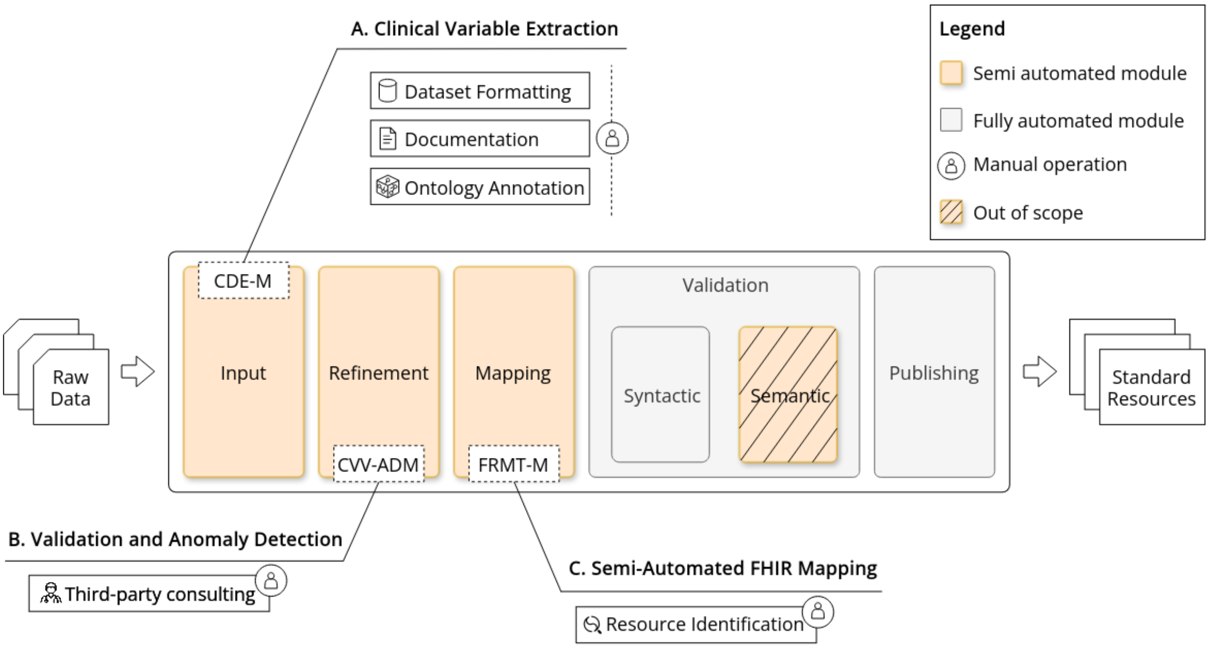

Figure 4: FLEX-LLM-CARE modular representation applied to the pipeline shown in Fig. 2. Modules highlighted in orange involve manual operations optimized through the proposed framework at each step: CDE-M for Input, CVV-ADM for Refinement, and FRMT-M for Mapping.

semantic search through the Facebook AI Similarity Search (FAISS) library, which allowed us to retrieve embeddings of standardized ontology terms (e.g., from SNOMED-CT and LOINC).

Subsequently, we focused on ensuring that the structured data was considered clinically consistent and semantically valid. To this end, we based the CVV-AD Module on ClinicalBERTand, above all, by implementing the related validation strategies (i.e., Guideline-Aware and Ontology-Driven). More precisely, for the Guideline-Aware, we employed ClinicalBERT to flag inconsistent entries, such as abnormal value ranges, contradictory clinical assertions, and missing dependencies. Instead, for the Ontology-Driven approach, we leveraged Apache Jena to convert validated data into Resource Description Framework (RDF) triples and utilized BioPortal to link them to various repositories of biomedical ontologies.

To streamline the mapping of validated data to the corresponding FHIR resources, we developed the FRMT module. This module leverages BioMistral-7B [32], guided by a few-shot learning approach. By providing illustrative examples—such as $(\texttt{ProstheticKnee}, \texttt{CommercialName}) \mapsto (\texttt{DeviceDefinition})$—the model adapts to the mapping logic with minimal supervision. To further improve accuracy and contextual relevance, we incorporated a RAG technique, similar to that used in the CDE module. This technique accesses a vector database populated with FHIR documentation, which supports BioMistral-7B during generation.

Finally, as described in Sec. 4.4, we enhanced the automation level of our modules using the 4D framework. Therefore, we used MLFlow to track our experiments, from the considered hyperparameters to the achieved performance metrics. However, all this has been made possible thanks to Hugging Face, which allowed us to download and employ the LLMs mentioned above. Additionally, we used LangChain to retrieve the required data (i.e., datasets and documentation) and support their conversion into corresponding embedding vectors.

5.2. Impact of FLEX-LLM-CARE Automation on Clinical Data Harmonization

We evaluated the performance of FLEX-LLM-CARE by applying it to the harmonization pipeline proposed by Marfoglia et al.[2]. Specifically, we assessed the reduction in manual effort by comparing the original process with the FLEX-LLM-CARE-enhanced pipeline. This evaluation was based on a mapping task involving a public dataset containing longitudinal clinical data on prosthetic patients [30].

Although the dataset being considered was already structured and well-documented, it lacked consistent semantic descriptions for many fields, making standardization and schema interpretation non-trivial. Additionally, data integrity checks were carried out using manually written Python scripts. These factors contributed to residual manual effort despite the dataset's relative maturity.

FLEX-LLM-CARE introduced automation in three modules—Input, Refinement, and Mapping—targeting the most labor-intensive steps: Schema interpretation, Data cleaning, Anomaly detection, and FHIR resource mapping. A first version of the FRMT module, leveraging BioMistral-7B, achieved approximately 60% accuracy in recommending appropriate FHIR resources for each source term. While this level of accuracy does not eliminate the need for manual review, it provides meaningful suggestions that significantly reduce expert time through partial matches, narrowed search spaces, and reduced cognitive load. Based on these benefits, we conservatively estimate a 60–65% reduction in mapping time.

Table 2 summarizes the estimated expert time required for each task in the original versus FLEX-LLM-CARE-assisted workflows. These estimates are based on historical project timelines [33] and expert feedback from the dataset curation process, during which four domain experts collectively contributed approximately 360 hours through a combination of collaborative sessions and individual annotation efforts. Task-level time allocations reflect this prior experience. While schema interpretation was relatively efficient due to the dataset's structured format, FHIR resource mapping remained the most demanding step, owing to the inherent complexity of aligning clinical terms with appropriate FHIR constructs.

Task	**Description**	**Pipeline Step [2]**	**FLEX-LLM-CARE Module**	**Manual (hrs)**	**Auto (hrs)**	**Saved (hrs)**
Schema interpretation	Interpret tables and define semantic roles.	Input	CDE-M	~40	~12	**~28 (70%)**
Data cleaning	Normalize formats, handle missing values.	Refinement	CVV-ADM	~15	~3	**~12 (80%)**
Anomaly detection	Identify implausible or inconsistent values.	Refinement	CVV-ADM	~15	~4	**~11 (73%)**
FHIR resource mapping	Select resources, define templates.	Mapping	FRMT-M	~250	~90	**~160 (64%)**
Syntactic validation	Ensure FHIR structural correctness.	Validation	—	~0	~0	—
Semantic validation	Conformance to FHIR profiles.	Validation	(Out of scope)	~40	~40	**0 (0%)**
Storage	Store and expose FHIR resources.	Publishing	—	~0	~0	—
Total				**~360**	**~149**	**~211 (59%)**

Table 2
Comparison of estimated expert time per task in the considered mapping process using the original pipeline [2] versus FLEX-LLM-CARE.

Despite the structured nature of the dataset, the automation introduced by FLEX-LLM-CARE led to significant time savings across all major stages of harmonization. The most substantial gains were observed in the *FHIR resource mapping* step, which was reduced by approximately 64%, primarily due to partial mapping assistance and structured suggestion triage. Overall, the framework reduced the estimated manual effort by approximately 211 hours, resulting in a 59% reduction in curation time.

These results underscore the potential of LLM-driven systems in enhancing the efficiency of clinical data harmonization, thereby making the adoption of th FHIR standard more accessible. While the reported time savings reflect the most tangible benefit, the integration of LLMOps through the 4D framework (see Sec. 4.4) also introduces critical operational advantages. These include robust experiment tracking, model versioning, and post-deployment monitoring, which collectively support reproducibility, transparency, and long-term maintainability. Although not directly reflected in the labor-hour estimates, these capabilities further strengthen the scalability and reliability of the FLEX-LLM-CARE approach, reinforcing its suitability for real-world observational research settings.

6. Conclusions

We introduced FLEX-LLM-CARE, a modular framework for clinical data standardization powered by LLMs and operationalized through the 4D LLMOps lifecycle. The framework includes three key modules: (1) clinical variable extraction from heterogeneous sources (CDE-M), (2) validation and anomaly detection (CVV-ADM), and (3) FHIR resource mapping through assisted selection (FRMT-M). Integrated into an existing pipeline and evaluated on a public dataset, FLEX-LLM-CARE can reduce manual curation effort by up to 59% while improving automation across key standardization stages.

A key insight is that coupling LLMs with structured operational support significantly lowers the manual burden of clinical data mapping while enhancing traceability, reproducibility, and maintainability. This combination enhances adaptability and reduces data integration overhead in complex environments, such as healthcare.

However, challenges remain, particularly in semantic validation, which was not deeply explored due to the computational cost of fine-tuning. Improving semantic precision remains difficult, especially when clinical relationships are subtle or implicit.

Looking forward, future work will explore: (a) enhancing retrieval-augmented generation (RAG) strategies to incorporate domain-specific knowledge better; (b) applying advanced embedding techniques to improve contextual understanding and output fidelity; and (c) benchmarking FLEX-LLM-CARE across diverse LLMs to assess robustness in real-world deployments.

As LLM ecosystems evolve, managing model versions, hyperparameters, and intermediate data will grow more complex. In this context, LLMOps will remain foundational, enabling dynamic experimentation, reproducibility, and safe deployment. Future work will continue to build on this operational backbone to advance scalable and semantically accurate FHIR harmonization.

Acknowledgments

This study was partially supported by the Italian Ministry of University and Research under PNRR-PNC Project PNC0000002 ”DARE – Digital Lifelong Prevention” (CUP: B53C22006450001).

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT, Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

[1] A. Chatterjee, N. Pahari, A. Prinz, HL7 FHIR with SNOMED-CT to Achieve Semantic and Structural Interoperability in Personal Health Data: A Proof-of-Concept Study, Sensors 22 (2022) 3756. doi:10.3390/s22103756.

[2] A. Marfoglia, F. Nardini, V. A. Arcobelli, S. Moscato, S. Mellone, A. Carbonaro, Towards real-world clinical data standardization: A modular FHIR-driven transformation pipeline to enhance

semantic interoperability in healthcare, Computers in Biology and Medicine 187 (2025) 109745. doi:10.1016/j.compbiomed.2025.109745.

[3] M. Lehne, J. Sass, A. Essenwanger, J. Schepers, S. Thun, Why digital medicine depends on interoperability, npj Digital Medicine 2 (2019) 79. doi:10.1038/s41746-019-0158-1.

[4] T. Benson, G. Grieve, Why Interoperability Is Hard, in: T. Benson, G. Grieve (Eds.), Principles of Health Interoperability: FHIR, HL7 and SNOMED CT, Health Information Technology Standards, Springer International Publisher, Cham, 2021, pp. 21–40. doi:10.1007/978-3-030-56883-2_2.

[5] European Commission, European Health Data Space Regulation (EHDS), https://health.ec.europa.eu/ehealth-digital-health-and-care/european-health-data-space-regulation-ehds_en, 2025.

[6] N. Pimenta, A. Chaves, R. Sousa, A. Abelha, H. Peixoto, Interoperability of Clinical Data through FHIR: A review, Procedia Computer Science (2023). doi:10.1016/j.procs.2023.03.115.

[7] J. Gehrmann, E. Herczog, S. Decker, O. Beyan, What prevents us from reusing medical real-world data in research, Scientific Data 10 (2023) 459. doi:10.1038/s41597-023-02361-2.

[8] R. Gazzarata, J. Almeida, L. Lindsköld, G. Cangioli, E. Gaeta, G. Fico, C. E. Chronaki, HL7 Fast Healthcare Interoperability Resources (HL7 FHIR) in digital healthcare ecosystems for chronic disease management: Scoping review, International Journal of Medical Informatics 189 (2024) 105507. doi:10.1016/j.ijmedinf.2024.105507.

[9] A. Carbonaro, A. Marfoglia, F. Nardini, S. Mellone, CONNECTED: Leveraging digital twins and personal knowledge graphs in healthcare digitalization, Frontiers in Digital Health 5 (2023). doi:10.3389/fdgth.2023.1322428.

[10] A. Marfoglia, C. D'Errico, F. Nardini, S. Mellone, A. Carbonaro, CONNECTED: A Knowledge Graph-Driven Platform for Clinical Data Harmonization and Personalized Digital Twin-Based Healthcare, in: 2025 IEEE International Conference (PerCom Workshops), IEEE Computer Society, 2025, pp. 116–121. doi:10.1109/PerComWorkshops65533.2025.00051.

[11] G. Lichtner, B. S. Alper, C. Jurth, C. Spies, M. Boeker, J. J. Meerpohl, F. von Dincklage, Representation of evidence-based clinical practice guideline recommendations on FHIR, Journal of Biomedical Informatics 139 (2023) 104305. doi:10.1016/j.jbi.2023.104305.

[12] N. Chen, J. Ren, An EHR Data Quality Evaluation Approach Based on Medical Knowledge and Text Matching, IRBM 44 (2023) 100782. doi:10.1016/j.irbm.2023.100782.

[13] N. Hong, A. Wen, F. Shen, S. Sohn, C. Wang, H. Liu, G. Jiang, Developing a scalable FHIR-based clinical data normalization pipeline for standardizing and integrating unstructured and structured electronic health record data, JAMIA Open 2 (2019). doi:10.1093/jamiaopen/ooz056.

[14] L. E. Lwakatare, A. Raj, I. Crnkovic, J. Bosch, H. H. Olsson, Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions, Information and Software Technology 127 (2020) 106368. doi:10.1016/j.infsof.2020.106368.

[15] S. Vänskä, K.-K. Kemell, T. Mikkonen, P. Abrahamsson, Continuous Software Engineering Practices in AI/ML Development Past the Narrow Lens of MLOps: Adoption Challenges, e-Informatica Software Engineering Journal 18 (2024) 240102. doi:10.37190/e-Inf240102.

[16] V. Moskalenko, V. Kharchenko, Resilience-aware MLOps for AI-based medical diagnostic system, Frontiers in Public Health 12 (2024). doi:10.3389/fpubh.2024.1342937.

[17] M. Reddy, B. Dattaprakash, S. Kammath, S. Kn, S. Manokaran, R. Be, Application of MLOps in Prediction of Lifestyle Diseases, ECS Transactions 107 (2022) 1191. doi:10.1149/10701.1191ecst.

[18] R. Shan, T. Shan, Enterprise LLMOps: Advancing Large Language Models Operations Practice, in: 2024 IEEE Cloud Summit, 2024, pp. 143–148. doi:10.1109/Cloud-Summit61220.2024.00030.

[19] J. Diaz-De-Arcaya, J. López-De-Armentia, R. Miñón, I. L. Ojanguren, A. I. Torre-Bastida, Large Language Model Operations (LLMOps): Definition, Challenges, and Lifecycle Management, in: 2024 9th International Conference on Smart and Sustainable Technologies (SpliTech), 2024, pp. 1–4. doi:10.23919/SpliTech61897.2024.10612341.

[20] A. M. Bennett, H. Ulrich, P. van Damme, J. Wiedekopf, A. E. W. Johnson, MIMIC-IV on FHIR: Converting a decade of in-patient data into an exchangeable, interoperable format, Journal of the

American Medical Informatics Association 30 (2023) 718–725. doi:10.1093/jamia/ocad002.

[21] M. Montazeri, R. Khajouei, A. Afraz, L. Ahmadian, A systematic review of data elements of computerized physician order entry (CPOE): Mapping the data to FHIR, Informatics for Health and Social Care 0 (2023) 1–18. doi:10.1080/17538157.2023.2255285.

[22] A. A. Sinaci, M. Gencturk, H. A. Teoman, G. B. Laleci Erturkmen, C. Alvarez-Romero, A. Martinez-Garcia, B. Poblador-Plou, J. Carmona-Pírez, M. Löbe, C. L. Parra-Calderon, A Data Transformation Methodology to Create Findable, Accessible, Interoperable, and Reusable Health Data: Software Design, Development, and Evaluation Study, J Med Internet Res 25 (2023). doi:10.2196/42822.

[23] F. Simon, J. Schladetzky, S. Macke, T. ABLAßa, J. Ingenerf, K.-S. Ann-Kristin, Metadata Driven Integration of Clinical Data for Secondary Use in FHIR-A Pilot Study at the UKSH, Studies in health technology and informatics 317 (2024) 146–151. doi:10.3233/SHTI240850.

[24] J. R. Jim, M. A. R. Talukder, P. Malakar, M. M. Kabir, K. Nur, M. F. Mridha, Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review, Natural Language Processing Journal 6 (2024). doi:10.1016/j.nlp.2024.100059.

[25] R. Garg, A. Gupta, A Systematic Review of NLP Applications in Clinical Healthcare: Advancement and Challenges, in: S. Das, S. Saha, C. A. Coello Coello, J. C. Bansal (Eds.), Advances in Data-Driven Computing and Intelligent Systems, 2024, pp. 31–44. doi:10.1007/978-981-99-9521-9_3.

[26] K. Klug, K. Beckh, D. Antweiler, N. Chakraborty, G. Baldini, K. Laue, R. Hosch, F. Nensa, M. Schuler, S. Giesselbach, From admission to discharge: A systematic review of clinical natural language processing along the patient journey, BMC Medical Informatics and Decision Making 24 (2024) 238. doi:10.1186/s12911-024-02641-w.

[27] S. Remmer, A. Lamproudis, H. Dalianis, Multi-label Diagnosis Classification of Swedish Discharge Summaries – ICD-10 Code Assignment Using KB-BERT, in: Proc. of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), 2021, pp. 1158–1166.

[28] M. A. Gulum, C. M. Trombley, M. Kantardzic, A Review of Explainable Deep Learning Cancer Detection Models in Medical Imaging, Applied Sciences 11 (2021). doi:10.3390/app11104573.

[29] S. Pahune, Z. Akhtar, Transitioning from MLOps to LLMOps: Navigating the Unique Challenges of Large Language Models, Information 16 (2025). doi:10.3390/info16020087.

[30] V. A. Arcobelli, S. Moscato, P. Palumbo, A. Marfoglia, F. Nardini, P. Randi, A. Davalli, A. Carbonaro, L. Chiari, S. Mellone, FHIR-standardized data collection on the clinical rehabilitation pathway of trans-femoral amputation patients, Scientific Data (2024) 806. doi:10.1038/s41597-024-03593-6.

[31] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: A pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2019) 1234–1240. doi:10.1093/bioinformatics/btz682.

[32] Y. Labrak, A. Bazoge, E. Morin, P.-A. Gourraud, M. Rouvier, R. Dufour, BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Findings of the Association for Computational Linguistics: ACL 2024, Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 5848–5864. doi:10.18653/v1/2024.findings-acl.348.

[33] V. Arcobelli, S. Moscato, A. Marfoglia, F. Nardini, P. Randi, A. Davalli, A. Carbonaro, P. Palumbo, L. Chiari, S. Mellone, MOTU on FHIR: A preliminary strategy to enable interoperability for retrospective dataset standardization, in: 2023 IEEE EMBS Special Topic Conference on Data Science and Engineering in Healthcare, Medicine and Biology, 2023, pp. 81–82. doi:10.1109/IEEECONF58974.2023.10404816.