Semantic Architecture for Interoperability in Distributed Healthcare Systems

EBTSAM ADEL\textsuperscript{1,2}, SHAKER EL-SAPPAGH\textsuperscript{3,4}, SHERIF BARAKAT\textsuperscript{1}, KYUNG SUP KWAK\textsuperscript{5}, (Life Senior Member, IEEE), AND MOHAMMED ELMOGY\textsuperscript{6}, (Senior Member, IEEE)

\textsuperscript{1}Department of Information Systems, Faculty of Computers and Information, Mansoura University, Mansoura 35516, Egypt
\textsuperscript{2}Department of Information Systems, Faculty of Computers and Information, Damanhour University, Damanhour 22511, Egypt
\textsuperscript{3}Faculty of Computer Science and Engineering, Galala University, Suez 435611, Egypt
\textsuperscript{4}Department of Information Systems, Faculty of Computers and Artificial Intelligence, Benha University, Elbahar 13518, Egypt
\textsuperscript{5}Department of Information and Communication Engineering, Inha University, Incheon 22212, South Korea
\textsuperscript{6}Department of Information Technology, Faculty of Computers and Information, Mansoura University, Mansoura 35516, Egypt

Corresponding authors: Kyung Sup Kwak (kskwak@inha.ac.kr) and Mohammed Elmogy (melmogy@mans.edu.eg)

This work was supported by the National Research Foundation of Korea through the Korean Government (Ministry of Science and ICT) under Grant NRF-2020R1A2B5B02002478.

ABSTRACT Electronic Health Records (EHRs) aggregate the entire patient’s data from different systems. Achieving interoperability for distributed EHR systems is expected to improve patient safety and care continuity, and therefore it improves the healthcare industry. However, achieving interoperability is challenging because of many standards, medical terminologies and ontologies, and different data formats. These formats make the integration of different systems an impossible process. If the hospital uses one standard to implement all of its medical systems, it will be no problem integrating them. However, hospitals usually depend on multiple standards and data formats to deliver different systems like hospital information systems, radiology information systems, laboratory information systems, etc. Semantic Web presents new technology for achieving EHRs interoperability. In this paper, we propose a novel ontological model to implement interoperability for distributed EHR environments. The proposed semantic ontology-based model can unify different EHRs data formats. In this study, We unify five different and popular healthcare data formats and standards. In addition, the framework could be extended straightforwardly to accept any other EHR data format. By implementing the proposed in real environments, we provide the physician with a single interface with a single terminology to query and interact with distributed healthcare systems that use different standards and data formats. This process is expected to help the physician to collect patient data from different systems quickly, completely, and correctly. The proposed ontological model has two stages. The first stage of the proposed converts each different input source to OWL ontology. In the second stage, it integrates all those ontologies into a merged crisp one. The integrated ontology includes 3753 axioms, 2606 logical axioms, 186 classes, 136 individuals, 126 datatype properties, and 257 object properties. We use SPARQL Protocol and RDF Query Language (SPARQL) and Description Logic (DL) queries to evaluate the output ontology. The obtained results ensure that the proposed framework helps physicians and specialists make a centralized point for all patients’ data. It could aggregate data with any heterogeneous structures with high precision.

INDEX TERMS Interoperability, electronic health record (EHR), ontology, openEHR archetype.
and flexible. Second, data in EHRs can be entered in a format and displayed in another format. Third, EHRs can integrate multimedia information as radiology images and echocardiographic videos. Finally, all authorized persons can arrive at patient data immediately when needed.

In 2009, Health Information Technology for Economic and Clinical Health (HITECH) was signed into law, represents the most extensive US initiative to date that is designed to encourage EHRs widespread use [3]. According to the World Health Organization (WHO) [4], the adoption of national EHR systems has been increased during the past 15 years. 50% of higher-income countries (for about 23), 35% of the middle-income countries (for about 10), 15% of the lower-income countries (for about 3) adopted national EHRs. Besides, EHRs deliver the right information to the right person at the right time using health Information Technology (IT). It has the potential to reduce up to 18% of patient safety errors and as many as 70% of adverse drug events [5].

Utilization of EHR reduces medical mistakes, increases revenue, and reduces costs. However, sharing the information needed is a very complicated problem. Doctors and specialists are suffering from collecting distributed data spread over different locations. The problem is that most practices are built on a heterogeneous mix of systems that may or may not be able to interface with one another and certainly can’t interface with systems outside the practice.

EHR could be achieved more efficiently by implementing syntactic “syntax” and semantic “meaning” interoperability. Many organizational standards were designed to achieve interoperability in the EHR environment, as was alluded in [6]. The evident reason for utilization standards is to standardize the way information is entered to the EHR. These standards are Digital Imaging and Communications in Medicine (DICOM) [7], Integrating the Healthcare Enterprise (IHE) [8], Clinical Data Interchange Standards Consortium (CDISC) [9], and Medical Markup Language (MML) [10]. There are also Health Level Seven (HL7) [11], CEN/ISO 13606 [12], and openEHR [13].

The Observational Medical Outcomes Partnership - Common Data Model (OMOP-CDM) and the openly Observational Health Data Sciences and Informatics (OHDSI) software initiatives are multi-stakeholder, international collaborators whose main objective is improving health by generating the evidence that promotes better care and better health decisions [14]. OMOP-CDM has the capacity to integrate data from different EHRs via an accepted data model.

Different healthcare providers should interoperate to improve the care quality [15], [16], [17]. ISO/TC 215 [18] defined Semantic Interoperability (SI) as “the capability of communicated systems to understand the shared information at the level of domain concepts formally defined.” Healthcare data by nature is more complex and has many parameters, which are growing quickly and in a distributed way. Also, the domain of healthcare produces a huge quantity of data from various disparate sources. The single patient’s data can be dispersed over diverse EHRs with various representation ways [19]. Data in EHRs could be presented in many different types of formats: structured as (database) [20], unstructured such as (documents, images, video, e-mails, and reports, and semi-structured data (XML files) [21]. Various complexities of healthcare systems can be found in [22] and [23].

Most literature studies are concerned with integrating specific data formats and standards that are not applicable in the real medical environment. Other literature is concerned with using one standard to handle the mentioned problem, but using a single standard isn’t sufficient to make unification, especially when the system is distributed. Semantic web technologies ease information integration of different systems [24]. This is because they can get the accurate meaning of the interchanged information. Ontologies have a significant role in solving the SI problem amongst many heterogeneous systems in distributed organizations, giving shared annotations and understanding for the common concepts [25]. According to the SemanticHEALTH project [26], EHR standards, ontologies, and terminologies are the main factors for accomplishing EHRs SI.

This paper is an expansion of our previous work [27]. Previously, we only proposed a semantic framework for achieving interoperability in distributed EHRs. The proposed framework depended on using ontology with the intention of its power in solving interoperability problems.

Ontology provides a vocabulary for a particular domain by describing its concepts, meanings, and relationships. Ontology enables sharing and reusing of health-related data. In addition, they make information both machine and human-readable. Ontology based on fuzzy logic enhances the crisp ontology’s capabilities and improves the medical domain’s accuracy by handling uncertainty problems. In this paper, we practically apply our proposed ontology-based framework and use benchmark datasets to conduct the experiments.

This paper implements the first two stages of the proposed framework. First, each input data source is represented in an ontology format. Then, all these output ontologies are integrated into a united ontology. The resulting ontology establishes a centralized point for all patients’ data. A specialist can ask any query and collect data from different locations regarding all the semantically and syntactically differences. Also, we execute complex semantic queries using SPARQL language to validate the proposed ontology. The main contributions of this paper are outlined in the following points:

1) It draws attention to the importance of SI in different EHRs formats. It enables healthcare physicians to access the required data in an integrated way. Thereby, it supports correct decisions at the correct time.

2) It invests semantic web capabilities for improving EHR management and achieves SI.

3) It proposes a semantically intelligent framework that has the ability to integrate different EHRs. The used data models are RDB, XML, Excel spreadsheet, CSV, and openEHR ADL archetype.
4) Different semantic queries are executed using SPARQL language to validate the proposed system.

The rest of this paper is structured as follows. Section II explains substantially some scientific introduction of the technologies utilized in the paper. Some related work and previous studies are manipulated in Section III. Section IV contains the used dataset description. The construction of the proposed ontology system with its internal phases are described in Section V. Section VI contains the results of experiments. We discuss our evaluation of the proposed solution in Section VII. Finally, the conclusion and future work are provided in Section VIII.

II. PRELIMINARIES

A. USED DATA MODELS

In this paper, we deal with many different EHRs data formats that could store medical data. These formats are relational databases (RDB), Extensible Markup Language (XML) documents, spreadsheets files, and openEHR Archetype Definition Language (ADL). The database is an aggregation of related data. Simply RDB could be defined by Eq. 1 as follows:

$$ R = (\text{Rel}(r), \text{Attr}(A, r), PK(p, r), FK(f, r)) $$

where Rel(r) is a relation in R, Attr(A) refers to A is an attribute in table T. PK(p) is a primary-key of T, and FK(f) is a foreign-key of T.

RDB provides the most successful technique for storing and retrieving data [20]. However, it suffers from weak semantics [28]. And thereby, there is an urgent need to represent the database semantically to be machine-readable [29], [30].

XML is a simple, easy-to-understand, and human-readable data format [21]. However all those advantages, it doesn’t support semantics nor reasoning [31]. That problem could be solved when converting XML documents into an ontology representation [31], [32].

Spreadsheets (Excel and CSV files) use a grid of cells coordinated in letter-named columns and numbered rows to regulate the stored data [33]. In the most recent years, public administrations and governments used to publish large amounts of structured data with tabular data formats, such as Excel spreadsheets or CSV files. They are characterized by simplicity and independence. Simply CSV file could be defined by Eq. 2 as follows:

$$ File_{csv} = (RC, FD, d, q) $$

where RC is a group of records, FD refers to a group of fields in the record, d is a delimiter character as comma, and q refers to an enclosing character.

OMOP-CDM enables capturing of patients’ information (e.g., patients, encounters, providers, drugs, diagnoses, procedures, and measurements) in the same way across different institutions. The OHDSI community has applied Natural Language Processing (NLP) to translate text into OMOP tables and fields. The purpose of a CDM is to standardize the format and content of observational data so that standardized applications, tools, and methods can be applied across different datasets. The use of a CDM integrates medical records across healthcare organizations so that these data resources can be queried to answer important questions efficiently and quickly. Standardized Structured Query Language (SQL) queries are shared in a common open-source repository.

B. EHR STANDARDS AND ARCHETYPES

Standard formats play a significant role in facilitating communication in EHR environments. However, each standard has a reference information model. The most common standards are HL7 [11], [34], openEHR [13], and CEN/ISO 13606 [12]. All these standards follow up the architecture of the dual model during representing clinical information. It relies on the separation between two distinct modeling levels, which are information (Reference model) and knowledge (Archetype model) [35]. The reference model describes the building blocks of the EHR information structure (as data types). The archetype model is utilized to constrain the information model structures. There are many software tools developed for managing archetypes, such as Clinical Knowledge Manager (openEHR CKM) [36], ADL Workbench (AW) [37], Archetype Editor (AE) [38], and LinkEHR [39].

The openEHR foundation [13] is a not-for-profit international company. It presents open specifications, software tools, and clinical models that can be utilized to create standards for solving interoperability in healthcare. It could be treated as a superset of CEN/TC 251 [12]. OpenEHR is relied on the archetype model in representing knowledge. An archetype can provide a standardized way of prescribing EHR hierarchies of clinical data and its data type [40]. Archetypes could be defined using ADL. It expresses the archetype as a text document [18]. ADL archetype includes three basic parts, which are header, definition, and ontology. The header section includes unique identifier data about archetype as name, version, author, and written language. The definition section includes a tree-structure restriction generated from the reference model. The ontology section contains constraints on terms and text, codes including the meaning of nodes, and connecting to terminologies like SNOMED-CT [41].

Clinical archetype has an essential role in achieving SI in distributed EHRs [42], [43]. It supplies a flexible building structure, which defines the information’s structure in terms of the taxonomic hierarchy. Its main idea is to give an interoperable and recyclable way for managing data validation and creation, while its reasoning and semantics are confined. However, it does not support rules or automated inference. Therefore, it was the point of departure for converting archetypes into an ontology representation.

C. SEMANTIC WEB

Tim Berners-Lee defined Semantic Web (SW) as “a set of best practices standards in sharing data and the semantics
of the shared data over the web for utilizing by different applications" [44]. Its main principle depends on using of semantic metadata. Semantic metadata gives information about the content of a document. Thereby it enables machines to comprehend the information semantics. SW supplies a widespread framework that allows data to be reused and shared via a different application, community boundaries, and enterprise. SW technologies have many capabilities, which achieve SI in any EHRs, such as [24]:

1) They give a shared annotation and understanding of the used concepts. It provides a uniform way to communicate between different parties in a distributed system and understand each other. And thereby, they enable the united utilization of distributed and heterogeneous contents.

2) They permit automated reasoning and deduce new information.

3) They permit the semantic representation and query of knowledge.

Ontology is considered the backbone of SW. Ontology is defined as a data model used to represent a set of concepts in a specific domain and relationships amongst these concepts [45]. It could be defined by Eq. 3 as follows:

\[
\text{Ontology} = (CL, CP, DP, OP, Dom(P), Rang(P))
\]

where CL represent the class, CP represent a class property, DP stands for a data type property, OP stands for an object property, Dom (P) represents the domain(s) of P, and Rang (P) represents its range(s).

Ontology has been built upon description logic (DL) language. Ontology alignment can be defined as a set of correspondences or relations (e.g., equivalence \(\equiv\), and subsumption \(\subseteq\)) between two ontologies [46]. Ontology is defined by a particular language. The Web Ontology Language OWL [47] is a W3C standard language that instantiates and defines SW ontologies. OWL is a general-purpose DL and is primarily used to describe classes of things in such a way as to support subsumption inference within the ontology, and by extension, on data that are instances of ontology classes. The subsumption \(C \subseteq D\) checks whether the first concept always denotes a subset of the second. DL has Boolean constructors as a conjunction (\(\land\)), which is interpreted as an intersection set, and negation (\(\neg\)) as a complement set, disjunction (\(\lor\)) as a union set. In addition to, it includes existential quantifier (\(\exists R.C\)), and value quantifier (VR.C). Table 1 shows OWL abstract syntax, the corresponding DL syntax, and its semantics.

Ontology composed of two main components: TBox (terminology) and ABox (assertions). TBox includes the building block on ontology such as: concepts (classes), individuals (instances), and roles (relationships), while ABox contains assertions about all those components [48]. Ontology provides a shareable, common, and reusable view of a particular domain [49]. It provides a formalized and meaningful way of constructing health data. Ontology is also used to reason about the objects within a specific domain.

### III. RELATED WORK

Through all the last years, many studies have tried to solve the integration and SI problem in distributed EHRs. In this section, we manipulate some of the studies tried to solve the mentioned problem. Santo and Medeiros [57] proposed a non-standard generic method that integrated a hierarchical organization of mediator schemas (mediators). The authors took knowledge bases to be the method for clinical data. This approach cloud query clinical data sources using semantic links across integrated knowledge bases.

El Hajjamy et al. [58] chose ontology as a common model to deal with the heterogeneity of different data models. Their framework was composed of three main phases. They provided a transparent and unique interface of classical data sources (RDB, UML, and XML) to local ontologies (OWL2). After that, syntactic similarity measurements amongst output local ontologies were calculated. Finally, all local ontologies were merged into a global ontological one. This technique avoided the output of ontology redundancy.

Roehrs et al. [59] proposed an interoperability model, “OmniPHR”. It presented a structural-semantic, unified, and up-to-date vision of PHR “personal health record” patients and healthcare providers. That model tried to achieve integration and SI between different health standards. It used a real with many adult patients’ medical records. The data was represented by three reference models: HL7 FHIR, openEHR, and MIMIC-III. The final execution score reached 87.9% F1-score.

Cristiano et al. [60] presented a method to reach the interoperability between heterogeneous health systems by using ontologies and rules. This method used the OWL features to map equivalences between mixed openEHR and HL7.

---

**TABLE 1.** OWL abstract syntax, corresponding DL syntax, and semantics Adopted from [48].

| OWL abstract syntax | DL syntax | Semantics |
|---------------------|-----------|-----------|
| Class(A partial C₁…Cₙ) | \(\neg A \subseteq \bigcap Cₖ \cap \bigcup Cₗ\) | A \(\equiv\) \(\bigcap Cₖ \cap \bigcup Cₗ\) |
| Class(A complete C₁…Cₙ) | \(A \equiv \bigcap Cₖ \cap \bigcup Cₗ\) | A \(\equiv\) \(\bigcap Cₖ \cap \bigcup Cₗ\) |
| SubClassOf(C₁ C₂) | \(C₂ \subseteq C₁\) | \(C₂ \subseteq C₁\) |
| EquivalentClasses(C₁…Cₙ) | \(C₁ \equiv \ldots \equiv Cₙ\) | \(C₁ \equiv \ldots \equiv Cₙ\) |
| DisjointClasses(C₁…Cₙ) | \(C₁ \cap C₂ \subseteq \perp\) | \(C₁ \cap C₂ \subseteq \perp\) |
| EnumeratedClasses(A a₁…aₖ) | \(A^1 \equiv \{a₁…aₖ\}\) | \(A^1 \equiv \{a₁…aₖ\}\) |

\(\sqcap\) (Subsumption), \(\sqcup\) (Intersection), \(\equiv\) (Equivalence), \(C:\text{Class}, A:\text{Attribute}\)
records. The used dataset was composed of heart rate and blood pressure readings of three patients extracted from the MIMIC-III database. They chose to represent each table's first five rows in openEHR while the rest were represented in HL7. Their approach translated HIS structure to OWL then created bindings between similar structures of each system. SWRL rules were used to increase the expressivity of the openEHR and HL7 ontology by classifying individuals based on their properties. These bindings are used by the reasoner to infer additional knowledge not explicit in the ontology.

Runumi Dev et al. [61] proposed An RDB to RDF mapping language approach that mapped patients’ RDB information maintained to an existing patients’ ontology. A mapping file was generated using R2RML for the customized and mapped database with the existing ontology. Finally, the RDF model was generated using R2RML processor based on the customized ontology. A simple SPARQL RDF Query Language query was executed that demonstrated ontology mapped patient’s triples.

Jaleel et al. [62] presented MeDIC, a framework of Medical Data Interoperability through Collaboration of healthcare devices. MeDIC improved over a cloud-based IoMT (Internet of Medical Things) by utilizing translation resources at the network edge, with its probing and translating agents. The probing agents preserve a capability list of MeDIC devices within a local network and enable one MeDIC device to request data conversion from another device when the former is not capable of this conversion by itself. The translation agent then converted the data into the required format and returned it to the former.

Figueiredo et al. [63] presented the BioFrame aiming to standardize and semantically annotate EHR forms. Their proposal was composed of a multi-dimensional conceptual model, an EHR specification process, and a catalog of software tools. The authors expected to contribute with a framework that enabled the efficient real-life application of semantic technologies and EHR standards to healthcare software systems. A case study was carried out on the specification of pediatric oncology nutrition records in a specialized hospital.

Sachdeva and Bhalla [64] aimed to discover knowledge representation for achieving semantic exchange of health data. OpenEHR and CEN TC251 EN 13606 use archetype-based technology. The authors presented various technologies for representing archetypes from XML to ADL to OWL through the KnowledgeRep simulation. Thus, the authors concluded that ADL is suitable for constraint representation and domain modeling. The OWL representation was the most suitable for semantic activities, without which the semantic exchange among the common standards was impossible. However, this study did not establish a comparison between distinct approaches (XML, OWL, OCL, and KIF) to interoperate within the healthcare domain. Furthermore, the researchers extended for various knowledge problems, such as insufficiency, incomplete and incorrect.

Jesus Moreno-Conde et al. [65] developed the MedicalForms system, which was based on the requirements identified in both the ISO/TS 13972 specifications and a Delphi study about tool requirements. The MedicalForms system proposed a consistent software-based methodology to support clinical information modeling providing support for terminology subset, information models and forms definition and validation. The system provided support for developing more than 100 structures and 50 forms in 15 projects related to research data collection and healthcare.

IV. DATASET DESCRIPTION
We obtain our experimental datasets from Mansoura University Hospital, Mansoura, Egypt. We discover that the whole hospital data are stored in one RDB. The EHR environment is usually distributed in many locations, and every location has a different data format and uses different terminologies. The integration of these sources is a challenge. As a result, in our experiments, we propose that patients’ data be stored in different formats, and we want to unify all of these formats. In the first model, we build a local patient database using MySQL-5.5.62. This database is developed to contain the main details of the patient and also his/her encounter history. The database contains the clinicians (5 columns & 5 test records), patients details (10 columns & 5 rows), diagnosis (3 columns & 5 records), medications (5 columns & 5 records), and encounter history (10 columns & 5 tuples) tables. The fields of each table with its datatype are shown in Figure 1.

The second model contains a blood pressure measurement Excel file with 17 records. It includes measurements of the patient’s blood pressure. Blood pressure could be defined as “the blood exerted force in the arteries when it circulates.” It is split into Diastolic Blood Pressure (DBP: when the heart is filling in) and Systolic Blood Pressure (SBP: when the heart is shrinking) pressures [66]. The Excel file contains the following fields: ROW_ID, PTIENT_ID, medicaid_ep_hospital_type, TESTDATE, SYSTOLIC (SBP), DIASTOLIC (DBP), and UNITS as described in Table 2. The third model is a CSV file contains 19 records, including some primary lab-tests for each patient. These tests are white blood cells, platelets, red blood cells, and hemoglobin. The file’s fields with their datatypes are illustrated in Table 3.

The fourth input model encompasses the patient’s vital signs, which are represented as an XML document. The upper part of Figure 8 shows a small part of the XML document created by XML Notepad. It contains some important patient’s vital signs, such as height, temperature, weight, oxygen saturation, respiratory rate, and smoking status. The final input model is an openEHR ADL archetype that involves a little part of colorectal screening, as shown in Figure 2. It is obtained from the Clinical Knowledge Manager (CKM) repository [36]. We use LinkEHR archetype editor [39] for editing archetypes.

The five input sources are entirely different and heterogeneous. The database is a structured model used for the storage
and management of data. The XML is a semi-structured model used for exchanging data. The Excel spreadsheets and CSV files can contain a high amount of organized data. The ADL is a standard way of storing patient data. Besides, the nature of medical data is more complex and has many parameters. Our prime motivation is to make unification between all heterogeneous sources. When a physician asks any query, EHRs should give an accurate answer regardless of all these semantically and syntactically distributed formats.

V. THE PROPOSED FRAMEWORK
The essential objective of this paper is to develop a semantically intelligent system for distributed EHRs. The roadway to achieving it is to build a unified single data model. This section describes our proposed ontology-based framework for EHRs interoperability. The architecture of this system is shown in Figure 3. It has two main layers: local ontologies construction and unified global ontology.
In the local ontologies construction “first layer,” each EHR data source is converted to an OWL ontology representation. The used input EHR data sources are RDB, spreadsheet files, XML documents, and openEHR ADL archetype. The output of this layer is a set of local ontologies for all input models. Every local ontology contains the same individuals represented in its corresponding model.

The proposed framework could be expanded to any other EHR data format. In this paper, we concentrate only on structured and semi-structured data. However, most EHRs data are stored in an unstructured format. In addition, unstructured clinical notes have evident value. At the same time, the main challenge of our proposed framework is to use unstructured data. Unstructured data should be processed somehow to be converted into unstructured or even semi-structured. And indeed, we might, in this case, depend on artificial intelligence technology.

In the global unified ontology “second layer,” all the output ontologies are merged into a global unified crisp ontology. The essential target of establishing a single global ontology is to give a unified and single user interface to all different and heterogeneous data sources. And thereby easily reducing all the required information from local ontologies through the combined one. This provides a clear meaning for the used concepts to high-level users. In ontology merging, all information about input ontologies components is preserved. There are various techniques and methodologies utilized in merging ontologies, such as PROMPT protégé plugin [67], VOAR (Visual and Integrated Ontology Alignment Environment) web-based [68], and ontology integration systems (OISs) [69]. The merged ontology must avoid redundancy and conflict between the same components. It maps an input entity in the first input ontology to an entity in the second input one.

**A. LOCAL ONTOLOGIES CONSTRUCTION**

This layer aims to construct a local ontological model from each input model. It could be suitable for any other format by following the appropriate conversion rules. In the following,
TABLE 4. Transformation from database to ontology.

| DR item            | Corresponding ontology element |
|--------------------|--------------------------------|
| DB table           | OWL Ontology class             |
| Tuple/Record       | Class instance/individual      |
| Field/Attribute    | OWL Data property              |
| Foreign key        | OWL Object property            |
| Field value        | Data property assertion        |
| Relationship       | Object property assertion      |
| Column constraints | Property cardinality           |

TABLE 5. Transformation from spreadsheet to ontology.

| Spreadsheet item    | Ontology construct |
|---------------------|--------------------|
| Spreadsheet file    | Ontology class     |
| Record              | Instance/individual|
| Field Header        | OWL Data property  |
| Field value         | Data property assertion |

we manipulate the mapping process for our experimental sources.

1) RELATIONAL DATABASE
The corresponding amongst RDB structure and the ontology elements are straightforward. The converting relational database to ontology has a set of sequential rules [70], [71], [72]. These rules describe how the components of RDB (containing rows, tables, constraints, columns, foreign keys, etc.) can be translated into ontology components (including classes, properties, axioms and instances). That process could be implemented automatically [70], semi-automatically [73], or manually. In the experimental assessment, RDB tables are represented as the OWL sub-class of the class Thing (it acts the set involving all individuals in the protégé project). Each tuple/record of a table is mapped to an instance of that table’s corresponding class. Each attribute is converted to be an instance of datatype properties. Each attribute will have the same data type as in the database. The value of this attribute is mapped as a value of the identical data type. Foreign keys are transformed to object properties. The mapping between database structure and ontology elements is illustrated in Table 4. The algorithm of mapping database to ontology is shown in Algorithm 1.

Algorithm 1 Transformation Relational Database Into Ontology

**Input:** Database Schema (S)

**Output:** Database Ontology (DBOnto)

```plaintext
/* Constructing the general structure of created ontology "TBox". */
Create empty Ontology ⊥ DBOnto /* Thing is the main class containing all sub-classes. */

for (Tb) in S.tables do
    OntoCls = CreateOntoClass(Tb.Name)
    OntoCls ⊑ ⊤
    FCL = Tb.FirstColumn

    if FCL is Primary Key (PK) then
        IDTP= CreateInverseFunctionalProperty(FCL.Type)
        OntoCls.AddDataTypeProperty(IDTP)
        IDTP.minCardinality = 1
    else if FCL is foreign key (FK) then
        OP = CreateObjectProperty(FCL.Name)
        OntoCls.AddObjectProperty(OP)
    else
        DTP= CreateDataTypeProperty(FCL.DataType)
        OntoCls.AddDataTypeProperty(DTP)
        (∀ OntoCls.DTP.DataType)
    end

    /* Fill the created ontology with the data “ABox”. */
    for Record (RD) in Tb do
       ClsInd= CreateClassIndividual(RD.Number)
       OntoCls.AddIndividual(ClsInd)
       for Column (CLV) in RD do
            DTPAss = CreateDatatypeAssertion(CLV.ActualValue)
            OntoCls.AddDatatypeAssertion(CLV)
        end

        /* Add the class to the created ontology */
        DBOnto.addClass(OntoCls)
    end

    /* Return the whole ontology that contains all the components of the schema. */
    Return DBOnto
```

2) SPREADSHEETS
CSV file is mapped to an ontology class. A record is mapped to class instances, headers are transformed to data properties, and the corresponding fields are mapped to the actual values. Algorithm 2 shows the Pseudocode of the process methodology. The mapping between spreadsheet file and ontology elements is illustrated in Table 5.

3) XML DOCUMENTS
XML data model is composed of node labeled tree, while the data model of OWL or RDF relies on triples of subject-predicate-object. So, mapping XML instances into OWL ontology is done easily and simultaneously. From the best of our knowledge and experience with all the tools used in converting XML to an ontology representation, the best way for converting XML documents to ontology is to generate a
Algorithm 2 The Methodology of Converting Spreadsheet to OWL Ontology

Input: Spreadsheet file SRDf
Output: Spreadsheet Ontology (SRDOnto)

/* Generating the building block of created ontology. */
Create empty Ontology ⊥ SRDOnto
SRDCls= CreateOntoClass(SRDf.Name)
SRDCls ⊑⊤ Row RD, Header HD, Field FD ← SRDf.Parse
FDTP= CreateDatatypeProperty(HD.Name)
SRDCls.AddDatatypeProperty(FDTP)
for Record (RD) in SRDf do
   ClsInd= CreateClassIndividual(RD.Number)
   SRDCls.AddIndividual(ClsInd)
   for Field (FD) in RD do
      FDTA= CreateDatatypeAssertion(FD.Name)
      SRDCls.AddDatatypeAssertion(FDTA)
   end
end
/* Add the class to the created ontology */
SRDOnto.addClass(SRDCls)
/* Return the whole ontology that contains all the components of SRDf. */
return SRDOnto

spreadsheet from XML. It is because XML exported from Excel spreadsheet is an XML oriented document. It describes the data structure and is utilized in encoding layout information. First, the XML file must be read as an excel file “through import from Developer menu” from Microsoft Excel. Then, we follow the same transformation rules as in spreadsheets.

4) openEHR ADL ARCHETYPES
We adopt ArchMS (Archetype Management System) to transform openEHR ADL archetype into an ontology format [24]. ArchMS is a platform for clinical data and archetypes management using SW technologies [74]. It allows importing ADL ISO 13606 and openEHR archetypes and represented them as OWL format. Also, it supports archetype management operations like search and comparison. It utilizes the services presented by Poseacle Converter [75]. When the ADL archetype is imported, it undergoes to the following of process.

1) Syntactic representation on ADL archetype by using:
   a) ADL parser: returns tree of AOM “Archetype Object Model” objects.
   b) XML serializer: takes AOM object tree and serialize it in XML schema.
   c) Eclipse Modelling Framework: it takes AOM XML schema as input and returns metamodel.

2) Transformation from syntactic model into semantic:
   a) Representing syntactic model into semantic one using MDE “Model Driven Engineering”.
   b) Using RubyTL to define corresponding transformation rules to obtain semantic archetype model.
   c) Obtaining OWL archetype.

Figure 4 clarifies transforming ADL archetype into OWL ontology ArchMS methodology.

B. GLOBAL UNIFIED ONTOLOGY
After constructing a local ontology for every input EHR model in the first phase of the framework, we aim to merge all these local ontologies into a global unified one. However, this step might include some conflicts. It might include redundancy between some entities. In addition, ontology could share concepts with the same synonymies “semantic”. In the experiment, PATIENT_ID is the only repeated entity. In Pro-tégé, each entity in the class has a unique IRI; we should rename IRIs of the same entities in both ontologies to be identical. The IRI of “Patients_details.PATIENT_ID” entity is the same IRI of “bloodpressuretest.PATIENT_ID” entity. The equation of semantic similarity between two concepts of ontology could be represented by Eq. 4.

\[
Sim(c_1, c_2) \equiv \frac{2H}{D_1 + D_2 + 2H} ; \quad (4)
\]

where \(c_1\) and \(c_2\) are two terms in concept C. D1 and D2 represent the shortest paths from \(c_1\) and \(c_2\) to C. H denotes the shortest path from c to the root. The Pseudocode of merging methodology is described in Algorithm 3.

VI. RESULTS
We perform the experiments to determine our proposed system’s effectiveness using heterogeneous formats, including RDBs, Excel spreadsheets, CSV, XML documents, and openEHR ADL archetype. This section discusses the gathered results for each phase of the proposed system. All the results are clarified with the help of screenshots.

A. HARDWARE AND SOFTWARE SPECIFICATIONS
Experiments were carried out on Inter Core(TM) i5-3360M CPU at 2.80GHZ 64 bit EliteBook with 8 GB of RAM and...
Algorithm 3 The Methodology of Merging Two Ontologies

Input: First ontology (Onto1) and Second ontology (Onto2). Output: Merged Ontology (ResOnto).

begin
/* Creating an empty merged ontology. */
Create empty Ontology ResOnto
for ∀ element ELM1 ∈ Onto1 do
  for ∀ element ELM2 ∈ Onto2 do
    /* Check similarities and differences between Onto1, Onto2 according to Eq. 4 */
    Check similarities between ELM1 and ELM2
    if ELM1=ELM2 then
      if ELM1 ∈ ResOnto then
        ResOnto.AddElement(ELM1)
        Neglect ELM2
      end
    end
  end
for ∀ element ELM2 ∈ Onto2 do
  for ∀ element ELM1 ∈ Onto1 do
    /* Check similarities and differences between Onto2, Onto1 according to Eq. 4 */
    Check similarities between ELM2 and ELM1
    if ELM2=ELM1 then
      if ELM2 ∈ ResOnto then
        ResOnto.AddElement(ELM2)
      end
    end
end
Return ResOnto
end

B. RESULTS OF LOCAL ONTOLOGIES CONSTRUCTION (FIRST PHASE)

In this section, we discuss the results of the first phase of the proposed ontological system (local ontologies construction). Each data source is converted into an ontology representation (RDF or OWL). With regards to MYSQL RDB, all the five tables were mapped into ontology classes with the same name of RDB. All 25 records were mapped into individuals. All 33 columns were mapped into Datatype Properties. All 4 foreign keys were mapped to Object Properties. The constructed ontology is shown in Figure 5. 493 logical axioms are generated during the process of converting RDB into ontology. Some of those generated Axioms are illustrated in Table 6.

A set of axioms could be defined to complete the precise semantics for each defined class. Axioms formulate the logical definitions of types. For example, (patients_details ⊆ Thing) denotes the concept of patients_details is a subclass from the parent class (Thing). We mapped all of the properties that could describe the patients_details subclass. For example, the formal description of (∀(∀ idpatient_details.Xsd:int) denotes each patient has at least idpatient_details of an int data type. Also, the formal description of (∀ (∃ encounter_history.ptient_id.encounter_history)) denotes each patient might have a previous visiting represented in encounter_history.ptient_id.encounter_history data property.

Concerning Excel format, the logical expression of (blood-pressuretest ⊆ Thing) denotes the concept of blood-pressuretest is a subclass from the main class. All seven columns were mapped into 7 Datatype Properties, and 17 records were successfully mapped to 17 individuals of the constructed ontology. Also, the logical description of (∀ (DIASTOLIC(DBP).Xsd:string)) denotes each patient record must have a string value for DBP measurement. The logical expression of (∀ (PATIENT_ID.Xsd:int)) denotes an integer value for each patientID. Figure 6 depicts the created ontology result from the conversation Excel file. One hundred ninety-two logical axioms are generated during these mappings. In addition, the blood-pressuretest class is subsumed by the Thing class as follows:

blood-pressuretest ⊆ Thing
∀ (DIASTOLIC(DBP).Xsd:string)
∀ (PATIENT_ID.Xsd:int)
∀ (ROW_ID.Xsd:int)
∀ (TESTDATE.Xsd:date)
∀ (SYSTOLIC(SBP).Xsd:string)
∀ (UNITS.Xsd:string)
∀ (medicaid_ep_hospital_type.Xsd:string)

Regarding the CSV spreadsheet, all 7 columns were successfully mapped to 7 Datatype Properties. 19 records were successfully mapped to 19 individuals of the constructed ontology. Figure 7 depicts the created ontology. 208 logical axioms are generated in this transformation. In addition, the Patientlabtests class is subsumed by the Thing class and some generated axioms are introduced as follows:

Patientlabtests ⊆ Thing
∀ (LABTEST.Xsd:string)
∀ (SUBJECT_ID.Xsd:int)
∀ (ROW_ID.Xsd:int)
∀ (DATETIME.Xsd:date)
∀ (NALUE.Xsd:string)
∀ (UNITS.Xsd:string)
∀ (NORMAL.Xsd:string)
Concerning XML format, the only main document element was mapped to ontology class, 2 sub-main elements were transformed into 2 individuals, and 8 document instances were successfully transformed into 8 Datatype Properties. 73 logical axioms are generated in this transformation. Figure 8 clarifies use-case scenario of constructing ontology from XML document. Some of the generated axioms are introduced as follows:

Patientvitalsigns ⊑ Thing
□ (∀ height.Xsd:string)

Concerning openEHR archetype, 1923 logical axioms are generated, 171 Classes, 68 individuals, 71 Datatype Properties, 250 Object Properties, and 66 Annotation.
Properties are generated. Part of mapping between openEHR ADL archetype and OWL ontology will be clarified in Figure 9.

Figure 10 lays out the constructed ontology of openEHR standard through OntoGraf. Some of the generated axioms are introduced as follows.
ARCHETYPE ⊑ Thing

ARCHETYPE_DESCRIPTION ⊑ Thing

ARCHETYPE_DESCRIPTION_ITEM ⊑ Thing

ARCHETYPE_TERM ⊑ Thing

ASSERTION ⊑ Thing

TERM_BINDING ⊑ Thing

TERM_DEFINITION ⊑ Thing

TERM_DATATYPE ⊑ Thing

TERM_BINDING

comment: Xsd:string

code_terminology: Xsd:string

description: Xsd:string

TERMINOLOGY ⊑ Thing

Table 7 summarizes the results of mapping different used data formats into ontology in terms of Classes, Object Properties, Axioms, Individuals, and Datatype Properties.

C. RESULTS OF ONTOLOGIES MERGING PROCESS (SECOND PHASE)

The integrated crisp ontology includes 2606 logical axioms, 186 classes, 136 individuals, 126 data properties, and 257 object properties. The class hierarchy and ontology metrics of the output united crisp ontology are depicted in Figure 11.

VII. DISCUSSION

Patients may have data distributed in many different locations, and the data have many different formats. The
distributed and heterogeneous data resources have different representations for the same data, which may lead to incorrect results. It is noticed that different systems are not able to communicate with each other as they don’t have a unified
TABLE 7. Results of local ontologies construction phase, in term of Axioms (AX), Classes (C), Datatype properties (DP), Object properties (OP), and individuals (I).

| Data source | Size (KB) | ontology (KB) | AX | C   | OP | DP | I   |
|-------------|-----------|---------------|----|-----|----|----|-----|
| MYSQL       | 22.7      | 76.9          | 493| 10  | 4  | 39 | 32  |
| RDB         | 1.16      | 29.5          | 192| 5   | 1  | 7  | 17  |
| CSV file    | 1.32      | 12.4          | 208| 4   | 1  | 7  | 19  |
| XML document| 706 bytes | 13.7          | 73 | 5   | 1  | 8  | 2   |
| openEHR     | 174 bytes | 62.1          | 1923|71|250|71|68   |

We run a set of queries and complex semantic queries on the obtained results. Queries in Protégé can be executed with the assistance of SPARQL [44] and DL Query [82]. The DL Query gives an interface for searching and ontology querying. The ontology has to be categorized before querying by a reasoner. We use Pellet reasoner [81] to validate integrated ontology consistency. Figure 12 depicts a simple SPARQL query that retrieves blood pressure measurements (from Excel file) and lab test value (from CSV file) for a specific patient. Figure 13 depicts a simple query that retrieves all patients (from patients’ details table), whose diagnosis was any type of cancer disease (from diagnosis table).

QUERY-1: Retrieve blood pressure measurements and lab test value for a specific patient.

QUERY-2: Retrieve all patients, whom the diagnosis was any type of cancer disease.

We evaluated our results by comparing the SPARQL results with the results of SQL server queries. The results were identical. There was neither duplication nor loss in the data obtained. As discussed in Section III, numerous techniques have been proposed to solve SI problems in distributed EHRs. However, most of these studies suffer from limitations that can be summarized in the following points.

1) Most literature is concerned with using one standard to handle the mentioned problem. However, in EHRs real implementation, each hospital could use its model or standard. The need to verify different data models in business domains is specifically highlighted. Therefore, this paper depends on using ontology to make unification amongst all EHRs distributed data models and standards.

2) Many studies proposed theoretical frameworks with neither implementations nor evaluation. This paper proposed, implemented, and evaluated a framework for overcoming the mentioned problems. The proposed framework uses ontology to make unification distributed EHRs in different models. In the near future, we will fuzzify the unified crisp ontology to handle the medical uncertainty problem.

3) Some studies are concerned with using one, two, or three data formats. However, most EHR systems currently available do not come in a one-size-fits-all format. EHR data could be physician-hosted systems or remotely-hosted systems. The new direction moves toward treating many different syntactic systems. This paper manipulates five different formats. Besides, our proposed framework could be widespread to include any other format.

Table 8 compares the proposed ontological model with the literature studies in terms of used data formats and interoperability methodology. Column 3 determines the manipulated EHRs data formats. Columns 4 refers to the adopted EHRs interoperability methodology. Column 5 points out whether EHRs standards are used. Column 6 demonstrates whether terminologies are used.

From the previous literature, we noted usage of a single standard isn’t sufficient to make unification, especially when the system is distributed. The new direction moves towards using a mediator as a classic strategy for interoperability when it has distributed data models. This work uses the ontology mapping mediator to convert every data source into...
TABLE 8. Comparison with the most interesting literature studies.

| Study                | Year | Data formats                  | Interoperability methodology | Standard | Terminology |
|----------------------|------|-------------------------------|-----------------------------|----------|-------------|
| El Hajjamy et al. [58]| 2018 | UML, XML, RDB                | Global crisp ontological     | No       | No          |
| Berges et al. [83]   | 2011 | RDB, ADL                      | Ontologies-mapping          | Yes      | Yes         |
| Kiourtis et al. [84] | 2017 | different EHR standard (e.g. HL7 v2, HL7 v3, ISO 13606, HL7 FHIR, etc.) | Ontology mapping            | Yes      | No          |
| da Costa et al. [60] | 2019 | MIMIC-III, openEHR, and HL7   | Ontologies + SWRL Rules     | Yes      | No          |
| Sartipi and Dehmoobad [85]| 2008 | HL7 v3, ACORD                   | Standard + Terminology      | Yes      | Yes         |
| Martínez-Costa et al. [86]| 2011 | openEHR and ISO/EN 13606     | Mediator ontology          | Yes      | No          |
| Santo and Medeiros [57]| 2017 | different DBMS                   | Mediator-based             | No       | Yes         |
| Shi et al. [87]      | 2010 | any DBMS                        | Mediation ontology        | Yes      | No          |
| El Azami et al. [88] | 2012 | different RDB                    | Fuzzy-based ontology      | No       | Yes         |
| Mylka et al. [89]    | 2014 | relational, XML and LDAP       | Ontology based             | No       | No          |
| Rochs et al. [59]    | 2018 | HL7 FHIR, openEHR, and MIMIC-III | Ontology based           | Yes      | No          |
| Korzun, and Meigal [90]| 2019 | Wireless personal area network | mobile multi-source data     | No       | No          |
| JALEEL et al. [62]   | 2020 | JSON, XML, Text           | Medical Data Interoperability through Collaboration of healthcare devices (MedIC) | No | No |
| Sachdeva, and Bhalla [64]| 2022 | XML and ADL archetypes            | Ontology-based             | Yes      | No          |
| Proposed             | -    | RDB, Excel, CSV, XML, and openEHR | Ontological-based        | Yes      | No          |

ontology representation. The output integrated ontology has the ability to make a centralized point for all patients’ data from heterogeneous EHRs data models. It fosters better health outcomes while reducing costs for healthcare providers and patients. However, many potential challenges and barriers could face its implementation in real health institutions. Many hospitals are still working using paper-based systems and have no EHRs. Other hospitals haven’t any EHRs standards. Using the EHRs has many current issues, such as privacy, security, and data quality. Before deploying the proposed work, the staff must be trained thoroughly about its workflow. There should be proper care coordination among multiple EHRs providers. Another issue is that the lack of information flow results in problems including inappropriate procedures; slower diagnosis and treatment.

Although the proposed framework achieves syntactic interoperability in distributed EHRs, it contains many limitations. Firstly, it had to handle an unstructured EHRs data model to increase our implementation scope. Unfortunately, most EHRs clinical data are unstructured and still not computable [91]. EHRs contain three main types of data structured (coded data: RDB, Lab tests, diagnosis codes), semi-structured (e.g., XML model), and unstructured information. Most of EHRs data is unstructured by nature (free-text clinical notes, radiology reports, and medical imaging, such as magnetic resonance imaging (MRI)). Structured and semi-structured formats are simple to retrieve, whereas unstructured data requires additional tools, such as natural language processing (NLP), to be retrieved. Handling unstructured data is an urgent issue for achieving successful and complete EHRs. Secondly, we need to create a graphical user interface for the implemented framework to be easy to use. Thirdly, we intend to measure the sensitivity of our proposed model. Fourthly, much of medical domain knowledge is vague, so we have to handle uncertain, incomplete, and vague problems in this domain.

VIII. CONCLUSION

In EHR, hospitals and doctors struggle to share the information needed to provide quality, timely, and cost-effective patient care. It turns out that data sharing in healthcare is much more difficult in execution than in concept. The problem is that most practices are built on a heterogeneous mix of systems that may or may not be able to interface with one another and certainly can’t interface with systems outside the practice. Our proposed ontological model has the ability to integrate and collect all patient data from heterogeneous data sources in a centralized point. It could improve care quality by making healthcare data available and accessible when and where needed and reducing medical errors. Combining multiple types of clinical data from the system’s health records has helped clinicians identify chronically ill patients. The proposed model is based on ontology SW technology. We used different EHRs data formats. These are as follows: MYSQL RDB, spreadsheet files (Excel and CSV), XML document, and openEHR ADL archetype. We converted each data format to an ontology representation (RDF or OWL). This phase can accommodate any other EHR data format.
Then, we integrated all the output ontologies into a unified crisp ontology. We used SPARQL query language to implement some queries. All the results were identical when compared with SQL server queries’ results. We expect our framework improves health care performance by achieving SI. Further development of the framework will concentrate on the limitations of this work, as discussed previously.

REFERENCES

[1] I. Iakovidis, “Towards personal health record: Current situation, obstacles and trends in implementation of electronic healthcare record in Europe,” Int. J. Med. Informat., vol. 52, nos. 1–3, pp. 105–115, 1998.

[2] J. Ehrenfeld and M. Cansenson, Monitoring Technologies in Acute Care Environments. New York, NY, USA: Springer, 2014.

[3] J. Adler-Milstein and A. K. Jha, “HITECH act drove large gains in hospital electronic health record adoption,” Health Affairs, vol. 36, no. 8, pp. 1416–1422, Aug. 2017.

[4] Electronic Health Records, World Health Org., Geneva, Switzerland, 2016.

[5] D. C. Kaelber and D. W. Bates, “Health information exchange and patient safety,” J. Biomed. Informat., vol. 40, no. 6, pp. S40–S45, Dec. 2007.

[6] E. Adel, S. El-Sappagh, S. Barakat, and M. Elmogy, “Distributed electronic health record based on semantic interoperability using fuzzy ontology: A survey,” Int. J. Comput. Appl., vol. 40, no. 4, pp. 233–241, Oct. 2018.

[7] DICOM. Digital Imaging and Communications in Medicine. Accessed: Mar. 1, 2021. [Online]. Available: https://www.dicomstandard.org/

[8] IHE (Integrating the Healthcare Enterprise). Accessed: Oct. 23, 2021. [Online]. Available: https://www.ihe.net/

[9] CDISC. CDISC (Clinical Data Interchange Standards Consortium). Accessed: Oct. 11, 2021. [Online]. Available: https://www.cdisc.org/

[10] J. Guo, K. Araki, K. Tanaka, J. Sato, M. Suzuki, A. Takada, T. Suzuki, Y. Nakashima, and H. Yoshihara, “The latest MML (medical markup language) version 2.3—XML-based standard for medical data exchange/storage,” J. Med. Syst., vol. 27, no. 4, pp. 357–366, 2003.

[11] Health Level Seven International (HL7). [Online]. Available: https://www.hl7.org/ accessed on 23, Sep. 2020.

[12] Health Informatics—Electronic Health Record Communication, Standard ISO 13606-1:2019. Accessed: Sep. 3, 2021. [Online]. Available: https://www.iso.org/standard/67668.html/

[13] OpenEHR. OpenEHR. Open Source Specifications, Models and Software for e-Health. Accessed: Sep. 30, 2020. [Online]. Available: https://openhehr.org/

[14] OHDSI. Observational Health Data Sciences and Informatics. Accessed: Apr. 23, 2022. [Online]. Available: https://www.ohdsi.org/

[15] I. Sommerville and T. Sawyer, Requirements Engineering: A Good Practice Guide, Hoboken, NJ, USA: Wiley, 1997.

[16] A. Rector, J. M. Rodrigues, D. Kalra, and P. Lewalle, “Semantic interoperability for better health and safer healthcare,” Eur. Commission, Directorate Gen. Inf. Soc. Media, 6th Framework Programme, Tech. Rep., 2009.

[17] L. DeNardis, “E-health standards and interoperability,” ITU-T Technol., Geneva, Switzerland, Watch Rep., Apr. 2012.

[18] P. Schloeffel, “Electronic health record definition, scope and context. ISO,” in Proc. TC, vol. 215, 2003, p.366.

[19] M. Dugas, P. Neuhaus, A. Meidt, J. Doods, M. Storck, P. Bruland, and J. Varghese, “Portal of medical data models: Information infrastructure for medical research and healthcare,” Database, vol. 2016, 2016, Art. no. bav121.

[20] R. Ramakrishnan and J. Gehrke, Database Management Systems. New York, NY, USA: McGraw-Hill, 2000.

[21] M. Seals, “The use of XML in healthcare information management,” J. Healthcare Inf. Manag., vol. 14, no. 2, pp. 85–96, 2000.

[22] P. E. Plesk and T. Greenhalgh, “The challenge of complexity in health care,” BMJ, vol. 323, no. 7313, pp. 625–628, 2001.

[23] S. E. Barakat Adel, S. El-Sappagh, and M. Elmogy, “A unified fuzzy ontology for distributed electronic health record semantic interoperability,” in U-Healthcare Monitoring Systems. Amsterdam, The Netherlands: Elsevier, 2019, pp. 353–395.

[24] M. D. C. Legaz-Garcia, C. Martinez-Costa, M. Menárez-Tortosa, and J. T. Fernández-Breis, “A semantic web based framework for the interoperability and exploitation of clinical models and EHR data,” Knowl.-Based Syst., vol. 105, no. 216, pp. 175–189, Aug. 2016.

[25] M. M. Taye, “Understanding semantic web and ontologies: Theory and applications,” J. Comput. Sci., vol. 2, pp. 182–192, Jun. 2010.

[26] S. Schulz and S. Schulz, “Semantic interoperability for health network deliverable 4.5 : Design guides and roadmap,” Version 2, European Commission, Brussels, Belgium, Tech. Rep., Jul. 2015.

[27] E. Adel, S. Barakat, and M. Elmogy, “Distributed electronic health record semantic interoperability based on a fuzzy ontology architecture,” in Proc. 14th Int. Conf. Comput. Eng. Syst. (ICCES), Dec. 2019, pp. 362–367.

[28] C. Martinez-Cruz, I. J. Blanco, and M. A. Vila, “Ontologies versus relational databases: Are they so different? A comparison,” Arifit. Intell. Rev., vol. 38, no. 4, pp. 271–290, Dec. 2012.

[29] K. D. Mogotlane and J. V. Fonou-Donbeu, “Automatic conversion of relational databases into ontologies : A comparative analysis of protege-2 plugs performances,” Int. J. Web Semantic Technol., vol. 7, no. 3/4, pp. 21–40, Oct. 2016.

[30] J. Varghese, S. H. Fenniri, O. Corcho, and D. P. Miranker, “Survey of directly mapping SQL databases to the semantic web,” Knowl. Eng. Rev., vol. 26, no. 4, pp. 445–486, Dec. 2011.

[31] N. Yahia, S. A. Mokhtar, and A. Ahmed, “Automatic generation of OWL ontology from XML data source,” 2012, arXiv:1206.0570.

[32] R. Ghawi and N. Cullot, “Building ontologies from XML data sources,” in Proc. 20th Int. Workshop Database Expert Syst. Appl., 2009, pp. 480–484.

[33] J. Mittloehner, S. Neumaier, J. Umbrich, and A. Polleres, “Characteristics of open data CSV files,” in Proc. 2nd Int. Conf. Open Big Data (OBD), Aug. 2016, pp. 72–79.

[34] T. Benson, Principles of Health Interoperability: SNOMED CT, HL7 and FHIR. London, U.K.: Springer-Verlag, 2016.

[35] T. Beale, “Archetypes: Constraint-based domain models for future-proof information systems,” in Proc. OOPSLA Workshop Behav. Semantics, vol. 105. Princeton, NJ, USA: Citeseer, 2002, pp. 1–69.

[36] Clinical Knowledge Manager (CKM). Apr. 30, 2021. [Online]. Available: https://openhehr.org/ckm/

[37] OpenEHR. ADL. Workbench. OpenEHR ADL Workbench (AWB) User Guide. Accessed: Jan. 2, 2022. [Online]. Available: http://openhehr.github.io/adl-tools/adl_workbench_guide.html

[38] OpenEHR Archetype Editor. Archetype Editor. Accessed: Sep. 2, 2021. [Online]. Available: https://www.openehr.org/downloads/archetypeeditor/home

[39] LinkEHR. LinkEHR Interoperability Platform. Accessed: Sep. 11, 2021. [Online]. Available: https://linkehr.veratech.es/

[40] A. Tapuria, D. Kalra, and S. Kobayashi, “Contributions of clinical archetypes, and the challenges, towards achieving semantic interoperability for EHRs,” Healthcare Informat. Res., vol. 19, no. 4, pp. 286–292, 2013.

[41] SNOMED CT. SNOMED International, Leading Healthcare Terminology, World Wide. Accessed: Sep. 11, 2021. [Online]. Available: https://www.snomed.org/snomed-ct

[42] E. T. Beale and S. Heard, “Architecture overview,” Univ. College, London, London, U.K., Tech. Rep., 2008, pp. 1–79.

[43] J. A. Maldonado, C. M. Costa, D. Moner, M. Menárez-Tortosa, D. Bosca, J. A. Miharro Giménez, J. T. Fernández-Breis, and M. Robles, “Using the ResearchEHR platform to facilitate the practical application of the EHR standards,” J. Biomed. Inf., vol. 45, no. 4, pp. 746–762, Aug. 2012.

[44] B. DuCharme, Learning SPARQL: Querying and Updating With SPARQL 1.1. Sebastopol, CA, USA: O’Reilly Media, 2013.

[45] N. Guarino, D. Oberle, and S. Staab, “What is an ontology?” in Handbook on Ontologies. Berlin, Germany: Springer-Verlag, 2009, pp. 1–17.

[46] F. Roda and E. Musulin, “An ontology-based framework to support intelligent data analysis of sensor measurements,” Expert Syst. Appl., vol. 41, no. 17, pp. 7914–7926, Dec. 2014.
SHAKER EL-SAPPAGH received the bachelor’s and master’s degrees in computer science from the Department of Information Systems, Faculty of Computers and Information, Cairo University, Egypt, in 1997 and 2007, respectively, and the Ph.D. degree in computer science from the Department of Information Systems, Faculty of Computers and Information, Mansoura University, Mansoura, Egypt, in 2015. In 2003, he joined as a Teaching Assistant at the Department of Information Systems, Faculty of Computers and Information, Minia University, Egypt. Since June 2016, he has been an Assistant Professor at the Department of Information Systems, Faculty of computers and Information, Benha University. From 2018 to 2020, he worked as a Research Professor at the UWB Wireless Communications Research Center, Department of Information and Communication Engineering, Inha University, South Korea. He is currently an Associate Professor with Galala University, Egypt. He is also a Senior Researcher with the College of Computing and Informatics, Sungkyunkwan University, South Korea. His publications in clinical decision support systems and semantic intelligence. His current research interests include machine learning, medical informatics, (fuzzy) ontology engineering, distributed and hybrid clinical decision support systems, semantic data modeling, fuzzy expert systems, and cloud computing. He is a reviewer of many journals and he is very interested in the diseases’ diagnosis and treatment research.

SHERIF BARAKAT received the B.Sc. and M.Sc. degrees from the Faculty of Science, Mansoura University, Mansoura, Egypt, and the Ph.D. degree from the Faculty of Science, Helwan University, Helwan, Egypt, in 2003. He is currently a Professor with the Department of Information Systems, Faculty of Computers and Information, Mansoura University. He has authored or coauthored over 50 research publications in peer-reviewed reputed journals, book chapters, and conference proceedings. His current research interests include machine learning, pattern recognition, and artificial intelligence.

KYUNG SUP KWAK (Life Senior Member, IEEE) received the Ph.D. degree from the University of California at San Diego, San Diego, CA, USA. He was with Hughes Network Systems, Germantown, MD, USA, and the IBM Network Analysis Center, Armonk, NY, USA. Since then, he has been with the School of Information and Communication Engineering, Inha University, Incheon, South Korea, as a Professor. He was the Dean of the Graduate School of Information Technology and Telecommunications. Since 2003, he has been the Director with the UWB Wireless Communications Research Center and the IT Research Center, Incheon. From 2006 to 2009, he was the President of the Korean Institute of Communication Sciences, Seoul, South Korea, and the Korea Institute of Intelligent Transport Systems, Seoul. His research interests include multiple access communication systems, mobile and UWB radio systems, the future IoT, and wireless body area networks. He is a member of the IEICE, KICS, and KIEE.

MOHAMMED ELMOGY (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Faculty of Engineering, Mansoura University, Mansoura, Egypt, and the Ph.D. degree from the Informatics Department, MIN Faculty, Hamburg University, Hamburg, Germany, in 2010. He is currently a Professor and the Chair of the Information Technology Department, Faculty of Computers and Information, Mansoura University. He worked as a Visiting Researcher with the Bioengineering Department, University of Louisville, Louisville, USA, from July 2016 to August 2019. He has authored or coauthored over 230 research publications in peer-reviewed reputed journals, book chapters, and conference proceedings. He advised more than 40 master’s and Ph.D. graduates. His current research interests include computer vision, medical image analysis, machine learning, pattern recognition, and biomedical engineering. He has served as a reviewer for various international journals. He served as a technical program committee member for many workshops and conferences.