Semantic Search Engine to Query into Electronic Health Records with a Multiple-Layer Query Language

Romain Lelong
Rouen University Hospital
1 rue de Germont
76000 Rouen France
romain.lelong@chu-rouen.fr

Chloé Cabot
Rouen University Hospital
1 rue de Germont
76000 Rouen France
chloe.cabot@chu-rouen.fr

Lina F. Soualmia
Rouen University Hospital
1 rue de Germont
76000 Rouen France
lina.soualmia@chu-rouen.fr

Stéfan J. Darmoni
Rouen University Hospital
1 rue de Germont
76000 Rouen France
stefan.darmoni@chu-rouen.fr

ABSTRACT

While the digitization of medical documents has greatly expanded during the past decade, health information retrieval has become a great challenge to address many issues in medical research. Information retrieval in electronic health records (EHRs) should also reduce the difficult tasks of manual information retrieval from records in paper format or computer. The aim of this article was to present the features of a semantic search engine implemented in EHRs. A flexible, scalable and entity-oriented query language tool is proposed. The program is designed to retrieve and visualize data which can support any Conceptual Data Model (CDM). The search engine deals with structured and unstructured data, for a sole patient from a caregiver perspective, and for a number of patients (e.g. epidemiology). Several types of queries on a test database containing 2,000 anonymized patients EHRs (i.e. approximately 200,000 records) were tested. These queries were able to accurately treat symbolic, textual, numerical and chronological data.

CCS Concepts

•**Information systems** → **Specialized information retrieval; Query representation;**

Keywords

Automatic Indexing, Electronic Health Record, Information Retrieval, Search Engine

1. INTRODUCTION

SIGIR '16 July 21, 2016, Pisa, Italy

DOI:

An Electronic Health Record (EHR) or an Electronic Medical Record is defined as *"an electronic version of the traditional record used by the healthcare provider"* [4]. EHR plays a central role since it includes a long-term record of care and a record of events from different types of care, including instructions, prospective information such as plans, orders and evaluations. In this context, the goal of the Information Retrieval (IR) System on EHR is to provide physicians with the correct information at the right place for the right person.

Several tools and frameworks for searching in EHRs for one patient have been proposed. These tools are adapted according to each data format: structured, not structured or mixed. CISearch has been developed and implemented in the Columbia University Hospital EHR. The CISearch end-user may query all the textual reports (imaging, pathology, discharge summaries, etc.), using certain Lucene tools. The objective of the LERUDI project was to perform IR from a projected French EHR model in emergency management, using a domain ontology. There are also various IR Systems (IRS) that are available based on health data warehouse for several patients. The main system is Informatics for Integrating Biology and the Bedside (I2B2), an open source platform developed in the USA and dedicated to translational research. I2B2 is one of the seven National Centers for Biomedical Computing funded by the National Institutes of Health. The I2B2 center focuses on developing a scalable informatics framework to bridge clinical research data with basic sciences research data. The framework uses coded data, biological data and other genomic data. The scope of the search concerns clinical search and statistical data analysis.

As described by Terry et al. [6], there are five basic options for searching specific data in EHR: (i) pre-determined queries: users select a query option from the software menu; (ii) simple customizable queries: users have some input into the queries to generate reports; (iii) advanced customizable queries: allow a greater amount of user input than the second level, often using Boolean logic; (iv) structured query language interface: using a special interface to enter Structured Query Language (SQL) commands; (v) data extract and analysis with database tools. Data semantics is particularly important as it derives from the concrete healthcare providing process in hospitals. EHR data is mainly composed of several key entities semantically related to one another: (a) patient, (b) hospital, (c) stay and then (d) the "classical" and more basic levels (procedures, diagnosis related group (DRG) coding, lab tests, reports, metadata from reports etc.). As a consequence, IR from EHR is more difficult and different when compared to the "classical" IR.

In this context, the aim of this study was twofold. First, describe a conceptual model which represents the conceptual and intuitive representation that non-IT medical provider users can have of EHR data. Secondly, describe a query language (QL) used to query those data and providing users the possibility to build queries accessing the entire set of EHR entities by taking advantage of the semantic network of entities. This search engine was funded by the French National Agency (TecSan program) in the Retrieval and Visualization In Electronic Health records (RAVEL) project.

2. MATERIALS

2.1 EHR Data Sources

A corpus of 2,000 anonymized patients and 200,000 reports from Rouen University Hospital (RUH) was used in this study, approved by the French National Commission on Computers and Liberty. Almost any clinical information available in the EHR is integrated in the RAVEL model, e.g. Diagnosis related group codes (ICD10), patient data (age, gender), lab tests and all medical reports. Moreover, natural language processing tools developed by the Vidal and Lille teams of the RAVEL project were also used to partially re-structure the unstructured data via multiterminological automatic indexing (AI) using more than 65 terminologies partially or totally translated into French.

2.2 EHR Conceptual schema and data model

The underlying database of the system is based on a generic Entity-Attribute-Value (EAV) physical data model [3]. This data model is able to integrate all types of data without structural changes to the data model (e.g. without adding columns or tables). This physical database structure enable to store any kinds of data in only a few tables. This helps to optimize IR, maintain the database and manage heterogeneous data types. As described in the thesis by A.D. Dirieh Dibad [2], EHRs are structured and organized in the four key concepts: (i) patient, (ii) hospital, (iii) stay, (iv) medical procedures, laboratory tests. A dedicated CDM using concepts i, iii and iv was designed to abstract the EHR data contained in the physical database data model (Figure 1). The query language syntax is patterned on that CDM instead of the physical database schema which provides the search engine with semantic features and capabilities.

3. METHODS

3.1 Query Language Description

The specific QL syntax is based on the CDM. Hence, building a query only requires real-life knowledge of existing entities in the database, their properties and their relationships with each other (e.g. *"a patient undergo a medical test"*). This query language has three main characteristics:

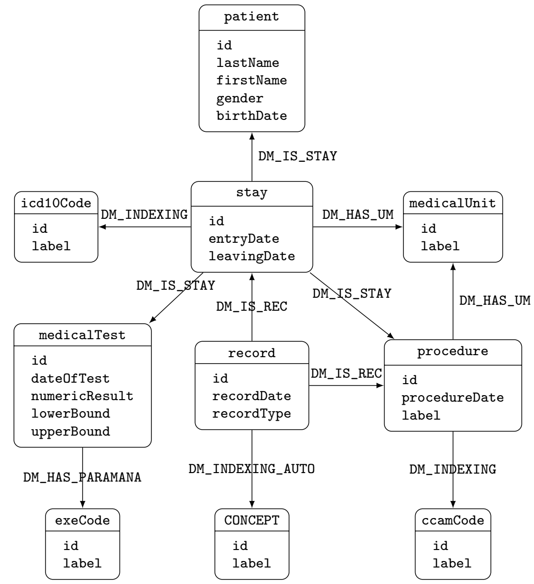

Figure 1: The entity-relationship diagram representing the EHR data

Semantic Information Retrieval capabilities: The QL is built with an entity-oriented vision. It enables semantic information retrieval since it provides the ability to display and query EHRs semantically related entities on any level (patient, stay, procedure, biological test etc.). It can also deal with multiple terminologies and hierarchical relationships.
Scalability: It is a scalable and flexible QL. It can automatically handle modifications on the CDM. More technically, new conceptual entities, attributes or relationships between entities are automatically taken into account directly from the database and without any search engine modification. This enabled an easy and rapid extension to omics data (genomics, metabolomics, proteomics, methylation etc.) [1].
Comprehensive querying capabilities: The full scope of entities can be queried using constraints built upon several types of data:
Textual and symbolic data (e.g. the query `patient(gender¬ ="M")` target male patients).
Numerical data (e.g. the query `medicalTest(numericResult > 6 AND numericResult <= 6.25)` targets lab tests with a result value between 6 and 6.25).
Chronological (eg. the query `stay(entryDate = 2010-03¬ -10)` targets stays, which entry date is `2010-03-10`).

All comparators and operators available are specified in Table 1.

3.2 Basic Querying

The query language is basically composed of nested syntactical units with the following syntax `ENTITY(CONSTRAINTS¬ _CLAUSE)`. `ENTITY` can correspond to any kind of entity of the CDM (e.g. `patient`, `stay`, `medicalUnit` etc.) and specify the type of object that the search engine should return (or target when nested). For instance, the queries `patient()` and `medicalUnit()` would respectively return all the patients and all the medical units of the database. The `CONSTRAINTS_CLAUSE` is a boolean expresion enabling to apply constraints to the targeted `ENTITY`. For instance, in the

Table 1: Types of data handled by the search engine

Data type	Available operators	Available comparators
character string data	None	`=` (equal), `!=` (not equal), `*` (wildcard)
Numerical data	`+` (add), `-` (subtract), `*` (multiply), `/` (divide)	`=`, `!=`, `<` (lower), `<=` (lower or equal), `>` (greater), `>=` (greater or equal)
Chronological data	`+`, `-`	`=`, `!=`, `<`, `<=`, `>`, `>=`

query `patient(birthDate=1937-01-01 AND gender = "¬M")` uses the two attributes `birthDate` and `gender` of the patient entity to return all male patients born on `1937-0¬1-01`. `stay(leavingDate-entryDate>=10)` will return stays with a duration of 10 days or more.

3.3 Semantic querying

The strength of the query language originates from its ability to deal with nested syntactical units. For instance the query `stay(patient(id = "DM_PAT_42"))` targets all stays associated with at least one relationship to the patient number 42. More complex queries can be performed by using the relationships between these entities. Some example queries are given in Table 2. This nesting functionality allows the exploitation of the relationships between entities and thereby enables to build queries based on the full semantic network. The QL has other querying capabilities: full text search, minimum and maximum on numerical data, hierarchical expansion, chronological and temporal queries.

3.4 Search Engine Process

The internal process of the search engine is composed of three main stages. Stages 1 and 2 are dedicated to build a comprehensive and computer-processable representation of the input string query. Stage 3 stands for the core of the search engine and consists of the precise querying of the EHRs data to return a list of entities.

Stage 1: Query parsing: A parser was designed to comprehensively define the query language syntax requirements. The parser matches and extracts this syntax respectively with and from the input. Finally, the parser enables to validate the structure of the query according to the query language specifications and split the query into several identified tokens corresponding to the elementary lexical and syntactic units of this query language.

Stage 2: Tree representation of the query: Stage 2 provides a computer-processable representation of a) the Boolean logic and b) the nested structure of the query. Stage 2 is particularly important as semantic search capabilities rely on it. A tree representation is an optimal computer processable structure to achieve that goal.

Stage 3: SQL query building: A SQL query is generated recursively from root nodes to leaf nodes of the tree built in Stage 2 and executed to return the list of entities.

4. RESULTS

4.1 RAVEL project use cases

Table 2: Examples of basic semantic querying

Query	`stay(patient(id="DM_PAT_1736") AND medicalUnit(label="Cardiology"))`
Description	All the patient `1736` stays which occur in the Cardiology medical unit.
Query	`stay(icd10SC(label="Burns involving less than 10% of body surface"))`
Description	All stays with a diagnosis of *"Burns involving less than 10% of body surface"* using ICD10
Query	`patient(medicalTest(exe(label="Sodium") AND numericResult>upperBound))`
Description	All patients linked (via stay's entity) to a biological test coded under `sodium` and with a result greater than normal i.e. all patients with hypernatremia.
Query	`medicalTest(medicalTest(label = "Sodium¬") AND numericResult<lowerBound AND patient(id="DM_PAT_125"))`
Description	For a given patient (number 125), display all hyponatremia test results.
Query	`patient(stay(icd10SC(id = "CIM_SC_T31.0¬") AND medicalTest(exe( label= "Sodium") AND numericResult>upperBound )))`
Description	All patients with a stay coded with the diagnosis *"Burns involving less than 10% of body surface"* (which correspond to the ICD10 sub category T31.0) and showing in that stay a hypernatremia.

Several use cases were successfully answered in the RAVEL project:

Use case 1: Visualize over time the neutrophil rate of a patient with rheumatoid arthritis

Use case 2: Produce all the medical reports containing the concept of metastasis

Use case 3: Retrieve all stays where *"REMICADE"* (inflix-imab) was used.

The use cases resolution required to use: AI in medical records, full text search, and multiple terminological ressources. Some of the queries used to answer these three use cases are shown in Table 3.

4.2 Comparison to I2B2 workbench

Table 3: Example of RAVEL search engine queries

Example	Description
`stay(patient(id="DM_PAT_21") AND procedure(label="BLOOD SAMPLE"))`	Patient 21 stays in which a blood sample procedure was token.
`medicalUnit(stay(patient(id¬="DM_PAT_21") AND procedure¬(label="BLOOD SAMPLE")))`	Medical units of the patient 21 stays in which a blood sample was taken.
`biologicalTest(patient¬(id="DM_PAT_1078") AND exe(label="Platelets") AND 10*numericResult<lowerbound)`	Patient 1078 platelet tests with a result more than 10 times lower than normal level.
`procedure(ccamMP(id="CCA_¬AM_EQQM006") AND procedureDate="MAX")`	The last procedure coded with EQQM006

Table 4: QL vs I2B2Functionalities

	QL	I2B2
Querying scope	1 or n entity	n patients
Querying	Textual query	Graphical querying
Detection of number of event occurrences	NO	YES
Lab test unit choice	NO	YES
Supported terminologies	69	14
Record AI	YES	NO
Omic data expression analyses for genes, proteins, micro-RNA and exons	YES	PARTIALLY

The I2B2 workbench is an open source patient cohort selection tool. The I2B2 workbench and the QL described in this study are both tools designed for searching in EHRs. However, the two tools have differences which are summarized in Table 4. The main difference is the querying scope. The I2B2 model focuses on the patient entity and is only designed for patient cohort constitution. In contrast, the search tool proposed in this study is more generic and can query any entity of the CDM. In fact, this tool can query n patients meeting precise criteria and search for any type of information (patients, lab tests, procedures, stays, etc.). This can be processed for one patient, for care, or for a set of patients, for epidemiology.

In the I2B2 workbench, the selection of a patient set is made through graphical user-friendly querying. Pre-defined constraints (e.g. age of the patient) can be dragged and dropped into a graphical query builder. This way of handling queries is more user-friendly but less flexible than the full textual query writing. Actually, writing plain queries is less easy to handle but requires less maintenance efforts and would be faster once mastered. Nevertheless, the I2B2 workbench provides numerous default features which cover a lot of use cases. It notably enables to detect the number of occurrences of an event contrary to the QL described in this study. The database on which the QL operates integrates currently 69 English and French terminologies which represent $2,340,655$ concepts partially translated into French. I2B2 workbench includes 13 terminologies (cf. Table 4) English for the major part. Other terminologies can be added. In contrast to I2B2 workbench, reports are automatically indexed and can be queried using the terminology terms with the QL. As regards cohort patient selection, I2B2 and the QL share most of their functionalities such as: numerical, chronological and textual constraints, full-text search on reports, search using concept subsumption, use of clinical data as constraints (stay data, medical unit data, patient data, etc.) and omic variant data management.

5. DISCUSSION

To date, the query language described in this paper is able to deal with levels 1 to 4 of Terry et al [6]. The global architecture of the underlying EHR system and the data querying strategy is closer to level 5 than to level 4 since, as reported by Terry et al [6] regarding level 5, the query language is based on the EHR's conceptual model. However, more advanced data analysis querying possibilities would probably be necessary to be considered as a full level 5 search options. Despite the fact the query language is quite complex to use, the public health professionals to whom it has been presented in fact stated that they would be able to use it after basic training. This training should also enable medical librarians, information scientists and IT specialists to use it. However, in contrast, several graphical user interfaces will be needed for health care professionals. These interfaces should provide access to more customizable queries than simple search. The I2B2 graphical interface could be a source of inspiration. To address this difficulty, an information extraction method was also designed in [5] to allow physicians to query EHRs using natural language instead of the dedicated QL. The search engine has been tested outside the Rouen University Hospital, Normandy: at Bordeaux University Hospital, Aquitaine, France. However, the current model still does not operate on the establishment level but should become operational in the near future. Furthermore, the comparative evaluation of this query language with I2B2 should be improved. A parser enabling to share data between I2B2 data model and the RAVEL data model could be implemented to accurately assess precision as well as querying scope of the query languages. A scaling up study is underway at Rouen University Hospital with all the patients with at least one stay (in or outpatient) in the dermatology department since 1992 (n=65,000). This study aims at querying EHR data in a multi-patient context in order to create a patient cohort.

6. CONCLUSION

In this study, a search tool dedicated to retrieving health information into an EHR has been presented. This search tool is able to adapt to any CDM and thus address a large variety of issues. Its specific query language provides practical and flexible querying capabilities but remains difficult to grasp for health professionals.

7. REFERENCES

[1] Chloé Cabot, Julien Grosjean, Romain Lelong, Arnaud Lefebvre, Thierry Lecroq, Lina F. Soualmia, and Stéfan J. Darmoni. Omic data modelling for informa on retrieval. In *2nd International Work-Conference on Bioinformatics and Biomedical Engineering*, pages 415–424, 2014.

[2] AD Dirieh Dibad. *Recherche d'Information Multi Terminologique au sein d'un Dossier Patient InformatisÃl'*. PhD thesis, University of Rouen, 2012.

[3] Prakash M Nadkarni. Qav: querying entity-attribute-value metadata in a biomedical database. *Computer methods and programs in biomedicine*, 53(2):93–103, 1997.

[4] Jeanne P Sewell and Linda Q Thede. *Informatics and nursing: Opportunities and challenges.* Wolters Kluwer Health/Lippincott Williams & Wilkins, 2013.

[5] Lina F Soualmia, Romain Lelong, Badisse Dahamna, and Stéfan J Darmoni. Rewriting natural language queries using patterns. In *Multimodal Retrieval in the Medical Domain*, pages 40–53. Springer, 2015.

[6] Amanda L Terry, Vijaya Chevendra, Amardeep Thind, Moira Stewart, J Neil Marshall, and Sonny Cejic. Using your electronic medical record for research: a primer for avoiding pitfalls. *Family Practice*, 27(1):121–126, 2010.