Definition of a Practical Taxonomy for Referencing Data Quality Problems in Health Care Databases

Paul Quindroit[1] Mathilde Fruchart[1] Samuel Degoul[2] Renaud Perichon[1] Niels Martignène[3,4] Julien Soula[1] Romaric Marcilly[1] Antoine Lamer[1,3,4]

[1] Univ. Lille, CHU Lille, ULR 2694 - METRICS: Évaluation des Technologies de Santé et des Pratiques Médicales, Lille, France
[2] Department of Anesthesiology and Intensive Care Unit, Groupe Hospitalier de la Région de Mulhouse et Sud-Alsace, Mulhouse, France
[3] F2RSM Psy - Fédération régionale de recherche en psychiatrie et santé mentale Hauts-de-France, Saint-André-Lez-Lille, France
[4] InterHop, Lille, France

Address for correspondence Paul Quindroit, PhD, CERIM, Faculté de Médecine, Pôle Recherche, 1 place Verdun, F–59045 Lille Cedex, France (e-mail: paul.quindroit@univ-lille.fr).

Methods Inf Med 2023;62:19–30.

Abstract

Introduction Health care information systems can generate and/or record huge volumes of data, some of which may be reused for research, clinical trials, or teaching. However, these databases can be affected by data quality problems; hence, an important step in the data reuse process consists in detecting and rectifying these issues. With a view to facilitating the assessment of data quality, we developed a taxonomy of data quality problems in operational databases.

Material We searched the literature for publications that mentioned "data quality problems," "data quality taxonomy," "data quality assessment," or "dirty data." The publications were then reviewed, compared, summarized, and structured using a bottom-up approach, to provide an operational taxonomy of data quality problems. The latter were illustrated with fictional examples (though based on reality) from clinical databases.

Results Twelve publications were selected, and 286 instances of data quality problems were identified and were classified according to six distinct levels of granularity. We used the classification defined by Oliveira et al to structure our taxonomy. The extracted items were grouped into 53 data quality problems.

Discussion This taxonomy facilitated the systematic assessment of data quality in databases by presenting the data's quality according to their granularity. The definition of this taxonomy is the first step in the data cleaning process. The subsequent steps include the definition of associated quality assessment methods and data cleaning methods.

Conclusion Our new taxonomy enabled the classification and illustration of 53 data quality problems found in hospital databases.

Keywords
- data quality
- database
- dirty data
- taxonomy
- data reuse

received
June 29, 2022
accepted after revision
November 2, 2022
accepted manuscript online
November 10, 2022
article published online
January 9, 2023

Georg Thieme Verlag KG,
Rüdigerstraße 14,
70469 Stuttgart, Germany

DOI https://doi.org/10.1055/a-1976-2371.
ISSN 0026-1270.

Introduction

In health care organizations, software packages and tools routinely generate and/or record huge volumes of data while they help users to perform their work. For example, software tools record the patients' stays in care units (for administrative purposes), laboratory test results (for optimizing diagnosis and treatment), and data from surgical theaters (to monitor the quality of care).[1]

In most hospitals, these operational applications have been implemented for several years now and may provide significant volumes of data of great value.[2] Indeed, several data reuse initiatives have been undertaken,[3–10] to discover new knowledge,[11] screen patients prospectively for inclusion in clinical trials,[12,13] provide physicians with teaching support,[14] and facilitate clinical research.[3–5]

However, the potential reuse of data is not always taken into account when databases are implemented and operated. For example, operational databases contain errors due to user input errors, poor documentation, measurement artifacts,[15–17] and inter-database differences in structure. As a result, the exploitation of these data can give erroneous results.[5,6] Data cleaning is one way of dealing with data quality problems.[5,7,12] Data cleaning typically comprises four main steps. First, it is mandatory to assess the quality of the source data; this assessment also provides an opportunity to judge the usefulness and accuracy of the software and the corresponding database. The various data quality problems can be related to the application's use, the database's design, the application's settings, and so on. Second, data cleaning processes are selected to address the detected data quality problems. Some data quality problems can be mastered and will not compromise the data reuse. Lastly, the data cleaning processes are implemented and the data are then re-evaluated (to measure the impact of cleaning).

To the best of our knowledge, a comprehensive taxonomy of data quality problems is currently lacking. Ideally, a taxonomy should (1) address all possible types of technical problems (i.e., from a single record to multiple data sources, including instances and structures), (2) systematically assess, manage, and improve data quality, and (3) facilitate the development of solutions that are quickly to implement and share.

Objectives

Here, we focused on the first step of the data cleaning process: the assessment of data quality. More precisely, we sought to classify technical problems and excluded data privacy, access, and security issues. The objectives of the present study were to define an operational taxonomy for data quality problems in operational health care databases, illustrate the taxonomy with concrete examples from clinical databases, and thus facilitate data quality assessments. To this end, we reviewed, summarized, and structured published works in this field.

Methods

For the sake of clarity and consistency, we applied the following names and definitions throughout the present manuscript. A database corresponds to a source of data (a data source or a source system) and is composed of several *tables* (also referred to as relations). A table stores *rows* (also referred to as tuples, lines, or records) characterized by various *columns* (attributes, fields, or variables). A value is stored in a cell at the intersection of a row and a column. Values in a given column can follow a predefined format (a syntax rule, grammar, or standardized format, e.g., "YYYY/MM/DD" for a date). A data quality problem is defined as *schema-related* when the problem arises from the data structure or, on the contrary, as *instance-related* when linked to the value itself (independently of its data type).

We applied a three-step method. The first step consisted in drawing up an inventory of data quality problems and their classifications, according to the literature. The second step consisted in structuring these problems in the most efficient way. Lastly, each data quality problem was illustrated with fictitious but realistic examples.

Review of Published Works

We searched the scientific literature for peer-reviewed English- or French-language publications containing a list or a taxonomy of data quality problems. The other main inclusion criterion was a practical definition and/or illustration for identifying data quality problems, preventing their occurrence, or lessening their impact.

The IEEE Xplore Digital Library, Springer, Science Direct, and MEDLINE via PubMed databases (time period: 1979–2022) were searched with terms "data quality problems," "data quality taxonomy," "data quality assessment," or "dirty data" (applied to the title, abstract, keywords, and/or full text). The search results (containing each publication's title, author(s), journal, and digital object identifier) were exported to an Excel file (Microsoft Corporation, Redmond, Washington, United States). Two reviewers (P.Q. and A.L.) independently checked the publication titles and abstracts against the selection criteria and then screened the selected full-text articles. Any discrepancies were resolved through discussion with S.D., R.P., J.S., R.M., N.M., and M.F., until a consensus was reached. The search was extended by searching in the references of the included documents.

For each publication, we extracted the taxonomy's objective, the data quality problems defined or illustrated, and the classification used to present these quality problems.

Organization and Implementation of the New Taxonomy

First, existing taxonomies were compared; we then chose the most comprehensive, intuitive taxonomy for the assessment of data quality in operational databases. Next, the data quality problems extracted from the reviewed publications were implemented in the chosen taxonomy. Data quality problems of the same type were grouped together. For example, a problem identified for a given field (e.g., missing

date of birth data) was grouped with other similar problems (e.g., missing weight data). Likewise, identical problems defined for different types of data were grouped together. We also identified similar data quality problems that occurred at different levels of granularity (e.g., violations concerning one tuple, many tuples, or many tables). When necessary, we harmonized the taxonomy's structure. For example, the same data problem could be defined with a positive or negative sentence (e.g., compliance or noncompliance with an integrity constraint); here, we chose to define the problem as the negative event.

Illustration of the New Taxonomy

The items selected for the new taxonomy were specified according to the classification. For the sake of clarity, each data quality problem was documented by a short title, a definition, and an illustration. Illustrations were created by transposing the defined problems to two realistic (but fictitious) hospital databases: an anesthesia information management system (AIMS) and an administrative software (ADMIN) that deals with hospital stays (steps, duration, diagnoses, and medical procedures). Simplified models of the data in these databases are shown in ►Fig. 1. Only tables storing facts (e.g., hospital stays and drug administrations) were selected; we omitted tables storing vocabularies (e.g., taxonomies of drugs, diagnoses, and medical acts). The AIMS database comprised five main tables: PATIENT (patient information), INTERVENTION (information on surgery), MEASUREMENT (monitoring), DRUG (drug administration), and EVENT (various events in anesthesia management). Two-dimensional tables were added to illustrate certain data quality problems. The ADMIN database also comprised five main tables: PAT (patient information), HOSPITAL_STAY (from admission to discharge), UNIT_STAY (details of stays in specific units), MEDICAL_PROCEDURE, and DIAGNOSIS. All the individuals portrayed in the illustrations were fictitious.

Results

The Literature Search

A total of 1,856 publications were identified in the initial literature search (IEEE Xplore Digital Library: $n = 410$; Springer: $n = 784$; Science Direct: $n = 259$; MEDLINE via PubMed: $n = 403$). We excluded 177 duplicate publications; hence, a total of 1,679 publication titles and abstracts were screened for relevance. After the first round of screening, 221 publications met our inclusion criteria. In the second round, 209 were excluded and so 12 publications were included. The main reasons for exclusion were a lack of data quality problems or publication in a language other than English or French. The selected publications' main characteristics are summarized in ►Table 1. Ultimately, 286 instances of data quality problems were extracted from the 12 publications. Although these instances were classified according to six distinct hierarchies in their original taxonomy, they were often similar, even if their wording was heterogeneous. The disparities in structure and in wording were mainly due to differences between the taxonomies' respective purposes (e.g., the evaluation of data quality tools).

Structure of the Taxonomy

We chose a taxonomic structure (►Fig. 2) based on the definitions given by Oliveira et al[18] and Rahm and Do.[17] This structure organizes data quality problems according to the corresponding database's levels of granularity. This approach makes it easier to review database objects and related problems occurring at the single column level, single row level, or multiple data source level. However, Oliveira et al focused on instance-related problems and excluded schema-related problems. Like Rahm and Do, we completed the structure by adding data quality problems related to the database's schema for each level of granularity.

The classifications implemented by Kim et al,[19] Li et al,[20] and Barateiro et al[21] referred to the root causes

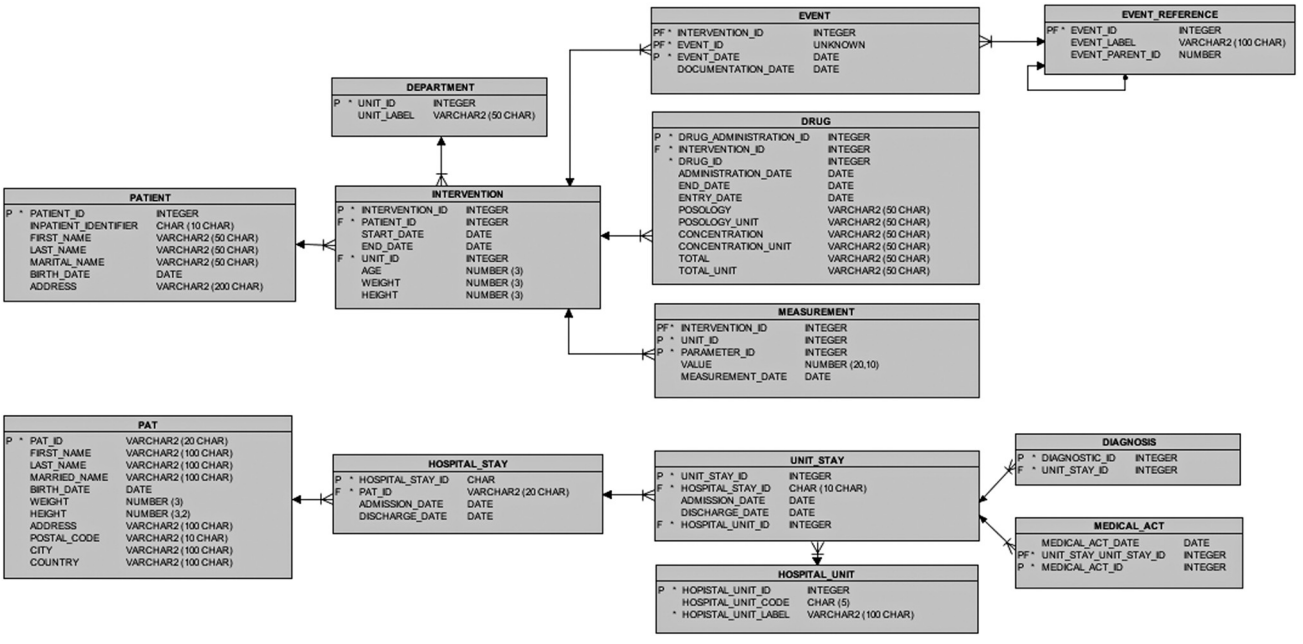

Fig. 1 The structure of the anesthesia information management system and administrative software applications used to provide examples.

Table 1 Publications containing a taxonomy of data quality problems, sorted by the date of publication

First author	Year of publication	Objective	Type of data quality problem classification presented in the publication	Number of items
Rahm	2000	To present data quality problems that cleaning methods have to address	Single source/multiple sources, and instance-/schema-related problems	18
Müller	2003	To present data quality problems that cleaning methods have to address	Syntactical/semantic/coverage anomalies	9
Kim	2003	To understand how dirty data arise and to determine which aspects have to be considered when cleaning the data	Hierarchical decomposition of three basic manifestations of dirty data: missing data, not missing but wrong data, and not missing and not wrong but unusable data.	33
Oliveira	2005	To evaluate and choose data quality tools, to guide research efforts	Granularity levels of databases	30
Barateiro	2005	To match categories of data quality tools with data quality problems	Schema level and content level, and then (1.1) avoided by database constraints (1.2) not avoided by database constraints (2.1) single record (2.2) multiple records	20
Li	2011	To detect dirty data	Rule-based taxonomy	38
Gschwandtner	2012	To add time-oriented considerations to existing taxonomies	Time-oriented taxonomy with a distinction between single-source and multiple-source problems.	25
Kahn	2016	To harmonize data quality terms into a comprehensive, unified terminology with definitions.	Dimension-based list (compliance, completeness, and plausibility)	16
Weiskopf	2017	To describe the formulation, development, and initial expert review of a data quality assessment.	Dimension-based taxonomy (conformance, completeness, plausibility, completeness, and currency)	9
Henley-Smith	2019	To establish a process for characterizing data quality, so that the quality assessment is tailored to the specifics of each intended secondary use.	Dimension-based taxonomy (conformance, completeness, and plausibility)	21
Wang	2020	To probe the potential benefit of data quality assessment and management.	Taxonomy without a hierarchy	28
Diaz-Garelli	*2022*	*To evaluate results and a standard prototype of a data quality assessment for cardiovascular disease risk assessment.*	*Taxonomy without a hierarchy*	*60*

of data quality problems (e.g., integrity through transaction management), the impact of quality problems on data reuse (e.g., documented, not-wrong data that were nevertheless not usable), and ways of avoiding data quality problems (e.g., by enforcing database constraints).[16,17,22] Gschwandtner et al present a taxonomy related to time-oriented data and that therefore failed to cover all the identified data quality problems.[23] Weiskopf et al, Wang et al, and Diaz-Garelli et al developed lists of data quality problems but did not rank or organize them.[24–26] Kahn et al and Henley-Smith et al organized data quality problems according to three dimensions of data quality: conformance, completeness, and plausibility.[27,28] We did not implement Müller's structure because it does not take into account of quality problems related to multiple sources.[29]

Elements of the Practical Taxonomy

After gathering together similar items from the 12 sources, the practical taxonomy comprised 53 items (►Table 3). For each category in the taxonomy, an example of data quality problem is fully documented below. In each example, the data quality problem is underlined.

Single Column of a Single Row

Data quality problem: missing value.
Definition: the value of a cell is null.
Example: in a row of the table PATIENT, the column BIRTH_DATE has a null value.

PATIENT (PATIENT_ID = 44908, INPATIENT_IDENTIFIER = "1001982736," FIRST_NAME = "JOSIANE," LAST_NAME = "DEWALLE," MARITAL_NAME = "ROSEY," BIRTH_DATE = "," …)

Sources:[17–21,23,24,29].

A Single Column in Multiple Rows

Data quality problem: unique value violation.
Definition: a column has the same value in different rows, whereas it is supposed to be unique.

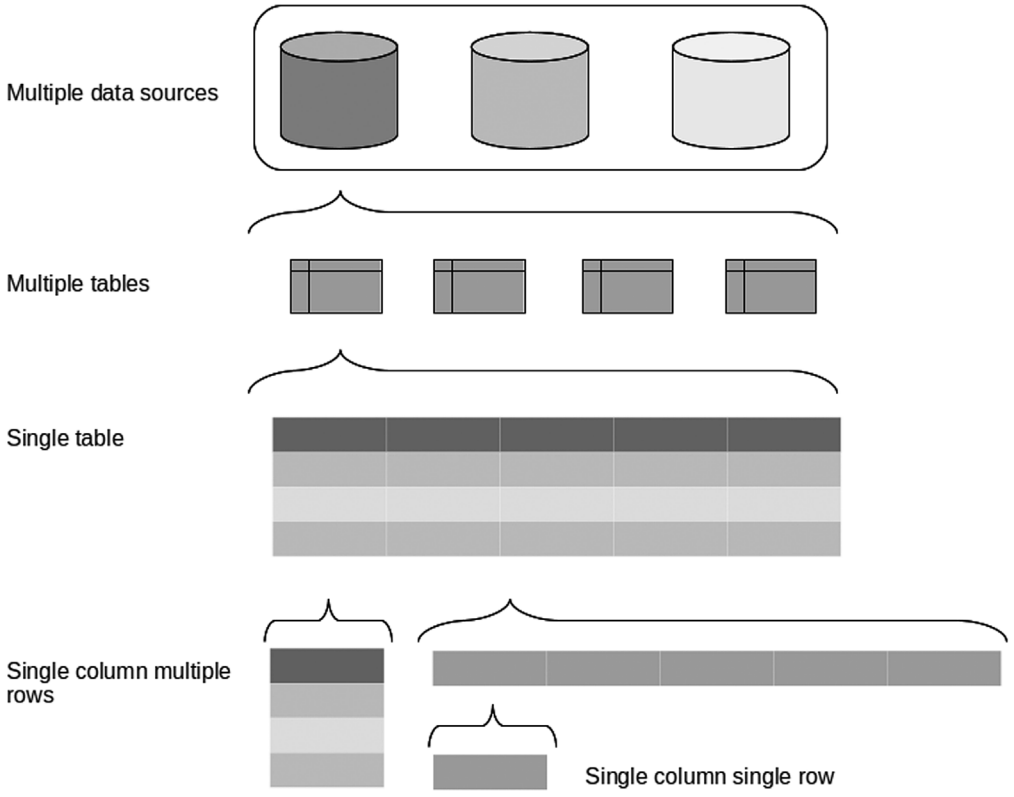

Fig. 2 The taxonomy's structure.

Table 2 Number of data quality problems per level of granularity

Acronym	Level of granularity	Number of data quality problems	
		Schema-related	Instance-related
SCSR	Single column of a single row	0	10
SCMR	Single column of multiple rows	1	4
MCSR	Multiple columns of a single row	0	4
ST	Single table	0	9
MT	Multiple tables	5	6
MDS	Multiple data sources	4	10

Example: in the PATIENT table, the following rows have the same inpatient identifier.

PATIENT (PATIENT_ID = 102310, INPATIENT_IDENTIFIER = "1002392301", ...)

PATIENT (PATIENT_ID = 104913, INPATIENT_IDENTIFIER = "1002392301", ...)

Sources:[17,27,28].

Multiple Columns in a Single Row

Data quality problem: wrong derived-field data.

Definition: a column calculated from other field columns shows an incorrect result.

Example: in a row of the DRUG table, the TOTAL column does not correspond to the product of the dosing frequency, the dose level, and the time period during which the drug is administered (expected value = 5).

DRUG(INTERVENTION_ID = 134454, DRUG_ID = 180, ADMINISTRATION_DATE = "2012/12/14 08:20:05," END_DATE = "2012/12/14 10:45:50," POSOLOGY = 2, POSOLOGY_UNIT = " mL/h," CONCENTRATION = 1, CONCENTRATION_UNIT = " mg/mL," TOTAL = 7, TOTAL_UNIT = "mg")

Sources:[19,20,23].

Single Table

Data quality problem: violation of a business domain constraint.

Definition: in a table, a row does not comply with a business domain constraint linked to another row.

Table 3 Taxonomy of data quality problems in operational databases

ID	Data quality problems	Definition	Example	Source	Schema/ instance
SCSR	*Single column of a single row*				
SCSR1	Missing value	The value of a cell is null.	In a row of the PATIENT table, the BIRTH_DATE column has a null value. PATIENT (PATIENT_ID = 44908, INPATIENT_IDENTIFIER = "1001982736," FIRST_NAME = "JOSIANE," LAST_NAME = "DEWALLE," MARITAL_NAME = "ROSEY," BIRTH_DATE = "", ...)	Barateiro, Diaz-Garelli, Gschwandtner, Kim, Li, Müller, Oliveira, Rahm	Instance
SCSR2	Dummy entry	The value of a cell corresponds to a default value, used by the source system or explicitly documented by the user.	In a row of the PATIENT table, the value of the BIRTH_DATE column is "01/01/1900," which corresponds to the default value for this field. PATIENT (PATIENT_ID = 54668, INPATIENT_IDENTIFIER = "1002982236," FIRST_NAME = "JEAN," LAST_NAME = "MEURISSE," MARITAL_NAME = "," BIRTH_DATE = "01/01/1900", ...)	Gschwandtner	Instance
SCSR3	Wrong data format	The data format does not comply with internal formatting constraints.	In a row of the PATIENT table, the value of the BIRTH_DATE column is documented in the format "MM/DD/YYYY" instead of the usual format "DD/MM/YYYY"; the patient's true birth date is June 1, 1996. PATIENT (PATIENT_ID = 125219, INPATIENT_IDENTIFIER = "1001082136," FIRST_NAME = "FRANCK," LAST_NAME = "AUSTER," MARITAL_NAME = "," BIRTH_DATE = "06/01/1996", ...) The zip code for France requires a 5-digits integer, whereas the value contains 4 digits only. In this example, the first digit "0" is missing. PATIENT (PATIENT_ID =230214, ..., POSTAL_CODE = "2843", ...)	Gschwandtner, Kahn, Kim, Li, Müller, Oliveira, Rahm	Instance
SCSR4	Invalid substring	A cell contains an invalid substring (e.g., special characters)	In a row of the PATIENT table, the LAST_NAME column also contains the substring "*09." PATIENT (PATIENT_ID =210320, INPATIENT_IDENTIFIER = "1001571146," FIRST_NAME = "JULIE," LAST_NAME = "BERTHE *09", MARITAL_NAME = "," BIRTH_DATE = "22/03/1978," ...)	Kim, Oliveira	Instance
SCSR5	Spelling mistake	A cell contains a spelling mistake.	In a row of the EVENT_REFERENCE table, a value is "Idnuction" instead of "Induction." EVENT_REFERENCE (EVENT_ID = 20198, EVENT_LABEL = "Idnuction", ...)	Barateiro, Kim, Li, Oliveira, Rahm	Instance
SCSR6	Imprecise value	The value of a cell is not complete enough to be interpreted (e.g., acronyms, abbreviations, lack of data elements, aliases, nicknames).	In a row of the PAT table, the POSTAL_CODE column contains the value "59," which is incomplete because the zip code must contain 5 digits to be useable. PATIENT (PATIENT_ID = 45908, ..., POSTAL_CODE = 59, ...)	Barateiro, Gschwandtner, Kim, Li, Müller, Oliveira, Rahm, Weiskopf	Instance
SCSR7	Embedded value	Multiple values are entered in the same cell.	In the following row of the PATIENT table, the values of the name and the marital name are both documented in the LAST_NAME column. PATIENT (PATIENT_ID = 101322, INPATIENT_IDENTIFIER = "1002371546," FIRST_NAME = "SIMONE," LAST_NAME = "DUBOIS / WIESPELINCK", MARITAL_NAME = "," ...)	Barateiro, Gschwandtner, Kim, Li, Rahm	Instance
SCSR8	Misfielded value	The value of a cell corresponds to the expected value of another cell.	In a row of the PAT table, the CITY column has the value "France" which should be entered in the COUNTRY column. PAT (PAT_ID = 2139923, ..., CITY = "France," COUNTRY = "")	Barateiro, Gschwandtner, Kim, Li, Rahm	Instance
SCSR9	Incorrect value	The value stored in a cell is not the true value, even though the value belongs to the domain.	In a row of the PATIENT table, the birth date of a patient is documented as "02/10/1980" whereas the patient was actually born on "20/10/1980." PATIENT (PATIENT_ID =230098, ..., BIRTH_DATE = "02/10/1980", ...)	Barateiro, Gschwandtner, Kim, Li, Müller, Oliveira, Wang, Weiskopf	Instance

Table 3 (*Continued*)

ID	Data quality problems	Definition	Example	Source	Schema/ instance
SCSR10	Domain violation	The value of a cell is outside the allowed range.	In a row of the DRUG table, the dosing frequency for a drug is a negative value but it should be greater than 0. DRUG (DRUG_ADMINISTRATION_ID = 1210938, INTERVENTION_ID = 210923, DRUG_ID = 23, ..., POSOLOGY = -2.5, ...)	Barateiro, Diaz-Garelli, Gschwandtner, Henley-Smith, Kahn, Kim, Li, Oliveira, Rahm, Wang, Weiskopf	Instance
SCMR	*Single column in multiple rows*				
SCMR1	Unique value violation	A column has the same value in several rows, whereas it is supposed to be unique.	In the PATIENT table, the following rows have the same inpatient identifier. PATIENT (PATIENT_ID = 102310, INPATIENT_IDENTIFIER = "1002392301", ...) PATIENT (PATIENT_ID = 104913, INPATIENT_IDENTIFIER = "1002392301", ...)	Henley-Smith, Kahn, Rahm	Instance
SCMR2	Different orders	Different orders are used in several rows, for a column containing concatenated data.	In the PATIENT table, the ADDRESS column contains data with different orders for the following rows: PATIENT (PATIENT_ID = 32910, ..., ADDRESS = "Downing Street, 32, London ") PATIENT (PATIENT_ID = 430103, ..., ADDRESS = "London, St Cross Street, 64 ")	Kim, Li	Instance
SCMR3	Existence of synonyms	In a column, some rows contain values that are synonyms or have the same meaning.	In the DRUG_REFERENCE table, two records correspond to the same drug: "Remifentanil" and "Ultiva." DRUG_REFERENCE (DRUG_ID = 134, DRUG_LABEL = "Remifentanil", ...) DRUG_REFERENCE (DRUG_ID = 1421, DRUG_LABEL = "Ultiva", ...)	Oliveira	Instance
SCMR4	Violation of a business domain constraint	Several values in a column do not comply with a business domain constraint.	In the EVENT table, the EVENT_DATE column (date) of the "end of surgery" event (EVENT_ID = 13) is earlier in time than EVENT_DATE column (date) of the "start of surgery" event (EVENT_ID = 12). EVENT (INTERVENTION_ID = 154872, EVENT_ID = 12, EVENT_DATE = "12/04/2010 12:20:43", ...) EVENT (INTERVENTION_ID = 154872, EVENT_ID = 13, EVENT_DATE = "12/04/2010 11:12:50", ...)	Henley-Smith, Oliveira	Instance
SCMR5	Wrong data type	The data type in a column is not constant across all rows.	In the INTERVENTION table, the AGE column has different formats, with rounding to one decimal place and rounding to an integer. INTERVENTION(..., AGE = 22.3, ...) INTERVENTION(..., AGE = 34, ...)	Barateiro, Diaz-Garelli, Gschwandtner, Kim, Li	Schema
MCSR	*Multiple columns of a single row*				
MCSR1	Semi-empty row	A row has some empty cells.	In a row of the PAT table, the columns related to the address (ADDRESS, POSTAL_CODE, CITY, COUNTRY) are empty. PATIENT (PAT_ID = 102908, LAST_NAME = "MARIETTE," MARITAL_NAME = "DEBEAUVAIS," FIRST_NAME = "JOSIANE," BIRTH_DATE = "01/08/1950," ADDRESS = "," POSTAL_CODE = "," CITY = "," COUNTRY = "")	Oliveira	Instance
MCSR2	Violation of functional dependency	In a row, a column does not comply with a functional dependency imposed by another column.	In the HOSPITAL_UNIT table, row 234 is incorrect because the unit code "3034" is documented as "Traumatology" instead of "Neurology" (the true value). HOSPITAL_UNIT(HOSPITAL_UNIT_ID = 234, HOSPITAL_UNIT_CODE = "3034," HOSPITAL_UNIT_LABEL = "Traumatology")	Oliveira, Rahm	Instance
MCSR3	Wrong derived-field data	A column calculated from other field columns shows an incorrect result.	In a row of the DRUG table, the TOTAL column does not correspond to the product of the dosing frequency, the dose level and the time period during which the drug is administered. DRUG(INTERVENTION_ID = 134454, DRUG_ID = 180, ADMINISTRATION_DATE = "2012/12/14 08:20:05," END_DATE = "2012/12/14 10:45:50," POSOLOGY = 2, POSOLOGY_UNIT = "mL/h," CONCENTRATION = 1, CONCENTRATION_UNIT = "mg/mL," TOTAL = 7, TOTAL_UNIT = "mg")	Kim, Gschwandtner, Li	Instance

(*Continued*)

Table 3 (*Continued*)

ID	Data quality problems	Definition	Example	Source	Schema/ instance
MCSR4	Violation of a business domain constraint	In a row, a column does not comply with a business constraint imposed by another column.	In the DRUG table, a business domain rule imposes the dosing frequency unit as a function of the administered drug. In line with this rule, administrations of ephedrine must be documented in milligrams. This rule is not complied with in the following row:	Henley-Smith, Kahn, Li, Oliveira, Wang	Instance
			DRUG(..., DRUG_ID = 123 (EPHEDRINE), ..., POSOLOGY = 10, POSOLOGY_UNIT = "ml", ...)		
ST	*Single table*				
ST1	Missing row	One row is missing, even though the information exists or the event has occurred.	In the EVENT table, the row corresponding to the "Start of surgery" event is missing, while the "End of the surgery" row is documented.	Gschwandtner, Kim, Li, Müller, Wang, Weiskopf	Instance
ST2	Unique value violation	Several columns have the same values over different rows, whereas they are supposed to be a unique combination of a primary key.	In the DRUG table, two records have the same combination for the INTERVENTION_ID, DRUG_ID and ADMINISTRATION_DATE columns, which compose the primary key.	Diaz-Garelli, Gschwandtner, Kahn, Kim, Li, Oliveira	Instance
			DRUG (DRUG_ADMINISTRATION_ID = 2013420, INTERVENTION_ID = 115310, DRUG_ID =234, ADMINISTRATION_DATE = "05/06/2012 10:34:21", ...)		
			DRUG (DRUG_ADMINISTRATION_ID = 2013422, INTERVENTION_ID = 115310, DRUG_ID =234, ADMINISTRATION_DATE = "05/06/2012 10:34:21", ...)		
ST3	Exact duplicate rows	Some rows in a table have identical column values (except for the primary key).	In the PATIENT table, two rows have identical values (except for the primary key).	Barateiro, Gschwandtner, Henley-Smith, Li, Wang	Instance
			PATIENT (PATIENT_ID = 483879, INPATIENT_IDENTIFIER = "1013420," LAST_NAME = "ELLOY," FIRST_NAME = "JUDE," BIRTH_DATE = "1966/04/22," ...)		
			PATIENT (PATIENT_ID = 393979, INPATIENT_IDENTIFIER = "1013420," LAST_NAME = "ELLOY," FIRST_NAME = "JUDE," BIRTH_DATE = "1966/04/22," ...)		
ST4	Approximate duplicate rows	Some rows of a table have identical column values (except for the primary key), while the values of some columns are greatly or slightly different.	In the PATIENT table, two rows have identical values for the INPATIENT_IDENTIFIER, LAST_NAME, FIRST_NAME, and BIRTH_DATE, columns but have distinct values for the CITY column.	Gschwandtner, Müller, Oliveira, Rahm, Wang	Instance
			PATIENT (PATIENT_ID = 23879, INPATIENT_IDENTIFIER = "120310293," LAST_NAME = "BOURGEOIS," FIRST_NAME = "CAROLINE," BIRTH_DATE = "1976/05/12," CITY = "PARIS")		
			PATIENT (PATIENT_ID = 34879, INPATIENT_IDENTIFIER = "120310293," LAST_NAME = "BOURGEOIS," FIRST_NAME = "CAROLINE," BIRTH_DATE = "1976/05/12," CITY = "SAINT-DENIS")		
ST5	Violation of a business domain constraint	In a table, a row does not comply with a business domain constraint toward another row.	In the EVENT table of AIMS1, two distinct (and mutually exclusive) ventilatory modes are documented for the same patient during the same period: spontaneous breathing (EVENT_ID = 158) and controlled ventilation (EVENT_ID = 159).	Diaz-Garelli, Gschwandtner, Henley-Smith, Kahn, Müller, Oliveira, Wang	Instance
			EVENT (INTERVENTION_ID = 250931, EVENT_ID = 158, EVENT_DATE = "2014/01/02 13:21:04",...)		
			EVENT (INTERVENTION_ID = 250931, EVENT_ID = 159, EVENT_DATE = "2014/01/02 13:21:04",...)		
ST6	Cyclic relationship problem	In a chain of associations, two rows reference each other through the foreign key column.	In the EVENT_REFERENCE table of the AIMS, there is a recursive relationship between events 24 and 50, through the EVENT_PARENT_ID column.	Li, Oliveira	Instance
			EVENT_REFERENCE (EVENT_ID = 24, EVENT_LABEL = " Induction," EVENT_PARENT_ID = 50)		
			EVENT_REFERENCE (EVENT_ID = 50, EVENT_LABEL = " Start of anesthesia," EVENT_PARENT_ID = 24)		
ST7	Outdated record	The data of a row are not up-to-date with the entity it defines.	In AIMS2, a record is meant to define the patient's current address rather than the patient's address at the time of admission.	Barateiro, Gschwandtner, Henley-Smith, Kim, Li, Weiskopf	Instance

Table 3 (*Continued*)

ID	Data quality problems	Definition	Example	Source	Schema/ instance
ST8	Unexpected result after aggregation	Row aggregation does not correspond to the expected result	In the AIMS 1, the number of anesthesia procedures per week shows an unexpected decrease.	Gschwandtner, Kahn	Instance
ST9	Unexpected variability	The values of two rows show unexpected variability for the same event of the same fact.	In the AIMS 1, different height values were found for the same patient.	Gschwandtner, Kahn	Instance
MT	*Multiple tables*				
MT1	Heterogeneous schema	Different schema representations of the same object in different tables.	In the MEASUREMENT and DRUG tables in the database 1, the units for measurements and drug administrations are represented differently. In the MEASUREMENT table, the units are documented with an identifier referencing the UNIT table; in the DRUG table, the dosing frequency, dose level, and total units are documented in a text field.	Barateiro	Schema
MT2	Homonyms in schema objects	The same name is used for different objects.	In the database 1, the term "UNIT" is used in the INTERVENTION and MEASUREMENT tables for the hospital units and the measurements units, respectively.	Barateiro	Schema
MT3	Synonyms in schema objects	Different names are used for the same object.	In the database 1, the EVENT and DRUG tables have different column names for the date of entry of a record: DOCUMENTATION_DATE and ENTRY_DATE, respectively.	Barateiro	Schema
MT4	Heterogeneous data types	The same real-world object is represented by different data types in different data sources.	In the EVENT_REFERENCES table of the database 1, the EVENT_PARENT_ID column references the EVENT_ID column but has two different types: INTEGER and NUMBER.		Schema
MT5	Heterogeneous data format	Two columns corresponding to the same real-world object have two different formats.	In the PAT table of the database 2, the BIRTH_DATE column has the format "YYYY/MM/DD" while the ADMISSION_DATE column of the HOSPITAL_STAY table has the format "DD/MM/YYYY."		Schema
MT6	Heterogeneous modalities	A categorical variable takes on different modalities in two different tables.	The smoking history is documented as YES/NO in the INTERVENTION table and as nonsmoker/smoker in the PATIENT table.		Instance
MT7	Heterogeneous units	Two columns corresponding to the same real-world object are expressed in different units.	The mean arterial pressure is documented in mmHg or cmHg, depending on the table (cmHg in INTERVENTION, and mmHg in MEASUREMENT).		Instance
MT8	Referential integrity violation	The value of a foreign key does not reference any of the rows in the primary key's table.	The UNIT_ID column in the INTERVENTION table references the row characterized by the primary key UNIT_ID = 215 from the UNIT table, while this row is missing in the UNIT table. UNIT (UNIT_ID = 213, ...) UNIT (UNIT_ID = 215, ...)	Barateiro, Gschwandtner, Henley-Smith, Kahn, Kim, Li, Müller, Oliveira, Rahm	Instance
MT9	Incorrect reference	The value of a foreign key references an existing value of the primary key, instead of another one.	In record 132760 of the INTERVENTION table, the UNIT_ID has a value of 153, whereas the operation was performed in unit 154. The two structures are present in the table STRUCTURE. INTERVENTION (INTERVENTION_ID = 132760, ..., UNIT_ID = 153, ...) STRUCTURE (UNIT_ID = 153, ...) STRUCTURE (UNIT_ID = 154, ...)	Gschwandtner, Li, Oliveira	Instance
MT10	Optionality relationship problem	The presence of a row in a table determines the presence of one or more rows in another table.	The operation 213420 does not have any records in the MEASUREMENT table, even though each operation is supposed to have measurements.	Li, Wang	Instance
MT11	Violation of business domain constraint	In a table, one or more rows do not comply with a business domain constraint toward a row from another table.	In the database 2 and for a given row of the HOSPITAL_STAY table, the UNIT_STAY table must contain related rows of admission dates and admission dates. The discharge date for the unit stay 100129902 is after the hospital discharge date.	Henley-Smith, Kahn, Kim, Li, Oliveira, Rahm, Wang, Weiskopf	Instance

(*Continued*)

Table 3 (*Continued*)

ID	Data quality problems	Definition	Example	Source	Schema/ instance
			HOSPITAL_STAY(HOSPITAL_STAY_ID = 3029180, ADMISSION_DATE = '2010/02/04', DISCHARGE_DATE = '2010/02/08')		
			UNIT_STAY(UNIT_STAY_ID= 100129283, HOSPITAL_STAY_ID = 3029180 ADMISSION_DATE = '2010/02/04', DISCHARGE_DATE = '2010/02/06')		
			UNIT_STAY(UNIT_STAY_ID= 100129902, HOSPITAL_STAY_ID = 3029180 ADMISSION_DATE = '2010/02/06', DISCHARGE_DATE = '2010/02/07')		
MDS	*Multiple data sources*				
MDS1	Heterogeneous schema	The same object has different schema representations in different databases.	In the database 1, information related to the address of the patient is stored in the ADDRESS column of the PATIENT table, while four distinct columns of the PAT table in the database 2 may store the ADDRESS, POSTAL_CODE, CITY, and COUNTRY.	Gschwandtner, Kahn, Müller, Rahm	Schema
MDS2	Homonyms in schema objects	The same name is used for different objects.	In the database 1 and 2, UNIT corresponds to a measurement unit and a hospital unit, respectively	Oliveira, Rahm	Schema
MDS3	Synonyms in schema objects	Different names are used for the same object.	In the databases 1 and 2, the patient object is represented by two distinct table names: PATIENT and PAT, respectively.	Oliveira	Schema
MDS4	Heterogeneous data format	Two columns corresponding to the same object comply with different regular expressions in different data sources.	In the database 1, the BIRTH_DATE column of the PATIENT table has the syntax DD/MM/YYYY, while in the database 2, the BIRTH_DATE column of the PAT table has the syntax YYYY/MM/DD.	Gschwandtner, Kim, Oliveira	Instance
MDS5	Heterogeneous data type	The same object is represented by different data types in different data sources.	In the database 1 and 2, the INPATIENT_IDENTIFIER column of the PATIENT and PAT tables is respectively named VARCHAR2(20) and CHAR(10).	Gschwandtner	Schema
MDS6	Heterogeneous modalities	A categorical variable takes on different modalities in different data databases.	In the database 1, the SEX column of the PATIENT table can have values of "0" or "1," whereas in the database 2, the SEX column of the PAT table can have values of "M" or "F."	Barateiro	Instance
MDS7	Heterogeneous units	Two columns corresponding to the same object are expressed in different units.	In the database 1, the HEIGHT column is expressed in meters in the INTERVENTION table, while in the database 2, this column is expressed in centimeters in the PAT table.	Gschwandtner, Kim, Li, Oliveira	Instance
MDS8	Heterogeneous encoding formats	Databases are encoded in different formats (e.g., ASCII, UTF-8).	The database is encoded with Latin-1 and the database 2 is encoded with UTF-8.	Li	Instance
MDS9	Approximate duplicate rows	In two databases, some rows about the same real-world object have identical column values (other than the primary key), while a few column values are greatly or slightly different.	In the tables PATIENT and PAT of databases 1 and 2, two records have identical values for the LAST_NAME, FIRST_NAME, and BIRTH_DATE columns but have different values for the CITY column.	Barateiro, Li, Oliveira	Instance
			PATIENT (PATIENT_ID = 23879, INPATIENT_IDENTIFIER = "120410195," FIRST_NAME = "CAROLINE," LAST_NAME = "BOURGEOIS," BIRTH_DATE = "1976/05/12," ADDRESS = "PARIS")		
			PAT (PAT_ID = 34879, FIRST_NAME = "CAROLINE," LAST_NAME = "BOURGEOIS," BIRTH_DATE = "1976/05/12," CITY = "SAINT-DENIS")		
MDS10	Inconsistent duplicate rows	In two sources about the same real-world object, one or more rows have identical identifiers while other columns have distinct values.	In the PATIENT and PAT tables of databases 1 and 2, two rows have identical values for the INPATIENT_IDENTIFIER but different values for the columns FIRST_NAME, LAST_NAME, BIRTH_DATE.	Barateiro, Li, Oliveira	Instance
			PATIENT (ID = 102879, INPATIENT_IDENTIFIER = "1039218273," FIRST_NAME = "DAMIEN," LAST_NAME = "FOUREST," BIRTH_DATE = "1956/11/12", ...)		
			PAT (PAT_ID = 106382,, INPATIENT_IDENTIFIER = "1039218273," FIRST_NAME = "ROMARIC," LAST_NAME = "LEJEUNE," BIRTH_DATE = "1946/05/19", ...)		

Table 3 (*Continued*)

ID	Data quality problems	Definition	Example	Source	Schema/ instance
MDS11	Violation of business domain constraint	In a database, a row does not comply with a business domain constraint toward a row from another data source.	An operation registered in database 1 should have a corresponding hospital stay in database 2. This is not the case for operation 132760, and the date of the operation is outside the dates of hospital stay. INTERVENTION (INTERVENTION_ID = 132760, START_DATE = "20/05/2010,") HOSPITAL_STAY (HOSPITAL_STAY_ID = 2301924, ADMISSION_DATE = "2010/05/22," DISCHARGE_DATE = "2010/05/25")	Oliveira	Instance
MDS12	Inconsistent aggregation	Data are aggregated with different denominators in different databases.	In the two databases, the number of patients per unit of time (e.g., week, month, or year) is different: database 1 stores patients with an anesthesia procedure, while database 2 stores all inpatients.	Li, Rahm	Instance
MDS13	Inconsistent timing	Records of the same entity in two different sources refer to different points in time.	An operation is documented as occurring on 02/03/2010 in the database 1 but is documented on 03/03/2010 in the database 2.	Rahm, Gschwandtner,	Instance
MDS14	Unexpected variability	Records for the same patient show inconsistencies between the two databases.	The smoking history is documented with unexpected variability in databases 1 and 2.	Kim	Instance

Example: in the EVENT table, the date of the "End of surgery" record is prior to the date of the "Start of surgery" record.

EVENT (INTERVENTION_ID = 250931, EVENT_ID = 158, EVENT_DATE = "2014/01/02 13:21:04,"…)
EVENT (INTERVENTION_ID = 250931, EVENT_ID = 159, EVENT_DATE = "2014/01/02 13:21:04,"…)

Sources:[18,23–25,27–29].

Multiple Tables

Data quality problem: referential integrity violation.
Definition: the value of a foreign key does not reference any rows in the table of the primary key.
Example: the column UNIT_ID in the INTERVENTION table references the row characterized by the primary key UNIT_ID = 215 from the STRUCTURE table, whereas this row is missing in the STRUCTURE table.

INTERVENTION (INTERVENTION_ID = 219321, …, UNIT_ID = 214, ...)
UNIT (UNIT_ID = 213, ...)
UNIT (UNIT_ID = 215, ...)

Sources:[17–21,23,27–29].

Multiple Data Sources

Data quality problem: synonyms in schema objects.
Definition: different names are used for the same object.
Example: in databases 1 and 2, the "patient" object is represented by two different table names, respectively PATIENT and PAT.
Source:[18].

Discussion

The objective of the present study was to define a practical taxonomy of technical data quality problems in operational databases. To this end, we reviewed the literature on published taxonomies. Instances documented in the selected taxonomies were gathered together and organized into a new, practical taxonomy. Each item of the new taxonomy was fully documented and illustrated with typical examples from operational health information systems.

By adopting a bottom-up approach, this taxonomy facilitates the systematic assessment of data quality in databases; it presents quality problems according to the data's granularity. In this way, exploration of the database structure can range from the most elementary structure (the value stored in a single cell) to more complex situations (data recorded by multiple sources). Moreover, we chose to combine schema- and instance-related problems in the same taxonomy. Lastly, each data quality problem was systematically illustrated with a (fictitious) example from a clinical database.

The main limitation of the new taxonomy is its scope; we focused solely on data quality problems in operational databases and did not consider data quality problems in data warehouses or after data cleaning.[30] Similarly, data quality problems related to authorization, accessibility, and security were not considered.[31] However, the taxonomy can be extended accordingly.

The items in our taxonomy are generic templates that must be implemented on the evaluated database, depending on the constituent tables and columns. For example, the *missing value* SCSR1 template could be instantiated with all columns for which a value is mandatory, as suggested by Diaz-Garelli et al and Wang et al.[24,25] Each data quality problem defined in the taxonomy could be completed with its incidence when assessing the data, as suggested by Henley-Smith et al.[27] Depending on the incidence and characteristics of the quality problems, one might also be able to give a criticality score for each data quality problem or an overall score for each data quality dimension, as defined by Weiskopf et al.[26]

Once the quality problem has been detected, the focus should be on its cause and potential measures for preventing

its occurrence in the source systems. It is useful to provide the software's developers and users with feedback during this step, to increase the data quality upstream of data storage.[30] The definition of assessment methods was outside the scope of the present study. Several data quality problems can be assessed by published automatic methods, whereas others always require manual analysis.[32] A further step in our research would be to link the data quality problems defined in our taxonomy to the appropriate assessment methods. Furthermore, data quality problems could be matched to the corresponding data cleaning methods. Lastly, we intend to assess our taxonomy with new data sources.

Conclusion

Based on the data quality problems reported in the literature, we defined a new taxonomy and illustrated it with 53 data quality problems from hospital databases. This taxonomy could be used to assess data quality problems during data reuse.

Conflict of Interest

None declared.

Acknowledgment

The authors thank David Fraser (Biotech Communication) for English editing.

References

1 Adler-Milstein J, Nong P. Early experiences with patient generated health data: health system and patient perspectives. J Am Med Inform Assoc 2019;26(10):952–959
2 Weng C, Kahn MG. Clinical research informatics for big data and precision medicine. Yearb Med Inform 2016;(01):211–218
3 Weiner MG, Embi PJ. Toward reuse of clinical data for research and quality improvement: the end of the beginning? Ann Intern Med 2009;151(05):359–360
4 Nunez CM. Advanced techniques for anesthesia data analysis. Seminars Anesthesia Perioperative Medicine and Pain 2004;23 (02):121–124
5 Prokosch HU, Ganslandt T. Perspectives for medical informatics. Reusing the electronic medical record for clinical research. Methods Inf Med 2009;48(01):38–44
6 Ebidia A, Mulder C, Tripp B, Morgan MW. Getting data out of the electronic patient record: critical steps in building a data warehouse for decision support. Proc AMIA Symp 1999:745–749
7 Safran C, Bloomrosen M, Hammond WE, et al; Expert Panel. Toward a national framework for the secondary use of health data: an American Medical Informatics Association White Paper. J Am Med Inform Assoc 2007;14(01):1–9
8 Meystre SM, Lovis C, Bürkle T, Tognola G, Budrionis A, Lehmann CU. Clinical data reuse or secondary use: current status and potential future progress. Yearb Med Inform 2017;26(01):38–52
9 McGlynn EA, Lieu TA, Durham ML, et al. Developing a data infrastructure for a learning health system: the PORTAL network. J Am Med Inform Assoc 2014;21(04):596–601
10 Chazard E, Ficheur G, Caron A, et al. Secondary use of healthcare structured data: the challenge of domain-knowledge based extraction of features. Stud Health Technol Inform 2018;255:15–19
11 Wade TD. Refining gold from existing data. Curr Opin Allergy Clin Immunol 2014;14(03):181–185
12 Miller JL. The EHR solution to clinical trial recruitment in physician groups. Health Manag Technol 2006;27(12):22–25
13 Cai T, Cai F, Dahal KP, et al. Improving the efficiency of clinical trial recruitment using an ensemble machine learning to assist with eligibility screening. ACR Open Rheumatol 2021;3(09):593–600
14 Altman M. The clinical data repository: a challenge to medical student education. J Am Med Inform Assoc 2007;14(06):697–699
15 Dentler K, ten Teije A, de Keizer N, Cornet R. Barriers to the reuse of routinely recorded clinical data: a field report. Stud Health Technol Inform 2013;192:313–317
16 Redman TC. The impact of poor data quality on the typical enterprise. Commun ACM 1998;41(02):x
17 Rahm E, Do H. Data cleaning: problems and current approaches. IEEE Data Eng Bull
18 Oliveira P, Rodrigues F, Henriques P. A formal definition of data quality problems. In:*ICIQ.*; 2005
19 Kim W, Choi BJ, Hong EK, Kim SK, Lee D. A taxonomy of dirty data. Data Min Knowl Discov 2003;7(01):81–99
20 Li L, Peng T, Kennedy J. A Rule Based Taxonomy of Dirty Data. In:2010. Doi: 10.5176/978-981-08-6308-1_D-035
21 Barateiro J, Galhardas H. A survey of data quality tools. Published online 2005. Accessed June 8, 2022 At: https://www.semanticscholar.org/paper/A-Survey-of-Data-Quality-Tools-Barateiro-Galhardas/1122bf09792b2cd93ef61d9fba24e2cbfd4e8325
22 Dasu T, Vesonder GT, Wright J. Data quality through knowledge engineering. In:*KDD '03.*; 2003. Doi: 10.1145/956750.956844
23 Gschwandtner T, Gärtner J, Aigner W, Miksch S. A taxonomy of dirty time-oriented data. In:*CD-ARES*; 2012. Doi: 10.1007/978-3-642-32498-7_5
24 Diaz-Garelli F, Long A, Bancks MP, Bertoni AG, Narayanan A, Wells BJ. Developing a data quality standard primer for cardiovascular risk assessment from electronic health record data using the DataGauge process. AMIA Annu Symp Proc AMIA Symp 2021; 2021:388–397
25 Wang Z, Talburt JR, Wu N, Dagtas S, Zozus MN. A rule-based data quality assessment system for electronic health record data. Appl Clin Inform 2020;11(04):622–634
26 Weiskopf NG, Bakken S, Hripcsak G, Weng C. A data quality assessment guideline for electronic health record data reuse. EGEMS (Wash DC) 2017;5(01):14
27 Henley-Smith S, Boyle D, Gray K. Improving a secondary use health data warehouse: proposing a multi-level data quality framework. EGEMS (Wash DC) 2019;7(01):38
28 Kahn MG, Callahan TJ, Barnard J, et al. A harmonized data quality assessment terminology and framework for the secondary use of electronic health record data. EGEMS (Wash DC) 2016;4(01):1244
29 Müller H, Freytag J. Problems, methods, and challenges in comprehensive data cleansing. Published online 2005. Accessed June 8, 2022 at: https://www.semanticscholar.org/paper/Problems-%2C-Methods-%2C-and-Challenges-in-Data-Mueller-Freytag/0168304c626a5b186bf559bf774a1dca52b04931
30 de Almeida WG, de Sousa RD, de Deus FD, Nze GDA, de Mendonça FLL. Taxonomy of data quality problems in multidimensional Data Warehouse models. Paper presented at: 8th Iberian Conference on Information Systems and Technologies; Lisbon, Portugal, June, 19–22, 2013
31 Strong D, Lee YW, Wang RY. Data quality in context. Commun ACM Published online 1997
32 Woodall P, Oberhofer M, Borek A. A classification of data quality assessment and improvement methods. Int J Inf Qual 2014;3(04): 298