Quality awareness for a Successful Big Data Exploitation

Cinzia Cappiello
Politecnico di Milano, DEIB
Milan, Italy
cinzia.cappiello@polimi.it

Walter Samá
Politecnico di Milano, DEIB
Milan, Italy
walter.sama@mail.polimi.it

Monica Vitali
Politecnico di Milano, DEIB
Milan, Italy
monica.vitali@polimi.it

ABSTRACT

The combination of data and technology is having a high impact on the way we live. The world is getting smarter thanks to the quantity of collected and analyzed data. However, it is necessary to consider that such amount of data is continuously increasing and it is necessary to deal with novel requirements related to variety, volume, velocity, and veracity issues. In this paper we focus on veracity that is related to the presence of uncertain or imprecise data: errors, missing or invalid data can compromise the usefulness of the collected values. In such a scenario, new methods and techniques able to evaluate the quality of the available data are needed. In fact, the literature provides many data quality assessment and improvement techniques, especially for structured data, but in the Big Data era new algorithms have to be designed. We aim to provide an overview of the issues and challenges related to Data Quality assessment in the Big Data scenario. We also propose a possible solution developed by considering a smart city case study and we describe the lessons learned in the design and implementation phases.

CCS CONCEPTS

• **Information systems** → **Uncertainty**; *Data model extensions*;

KEYWORDS

Data Quality Assessment, Big Data, Veracity

ACM Reference Format:
Cinzia Cappiello, Walter Samá, and Monica Vitali. 2018. Quality awareness for a Successful Big Data Exploitation. In *IDEAS 2018: 22nd International Database Engineering & Applications Symposium, June 18–20, 2018, Villa San Giovanni, Italy*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3216122.3216124

1 INTRODUCTION

The management of different scenarios (e.g., city administration, public transportation, tourism, retail industry) can be improved by taking advantage of the big amount of collected data. Big Data sources indeed enable advanced analysis that might reveal the real status of actual systems and the feasibility of improvement actions (i.e., new solutions or modifications to the current infrastructure). In particular, researches in this field try to acquire new knowledge by seeking patterns, formulating theories and testing hypothesis using data gathered from different sources. The problem is that not all the data are relevant: "one of the fundamental difficulties is that extracted information can be biased, noisy, outdated, incorrect, misleading and thus unreliable" [3]. For this reason it is important to assess quality of data before using them in order to take valuable strategic decisions. In fact, an analysis based on noisy or incomplete information can generate wrong and problematic results.

Data Quality (DQ) is often defined as "fitness for use", i.e., the ability of a data collection to meet users' requirements [13]. It is evaluated by means of different quality dimensions. The dimensions that are relevant in most of the studies are accuracy, completeness, timeliness, and consistency [2]. The evaluation of such dimensions is a way to address veracity but Big Data pose new challenges: (i) volume increases the complexity of Data Quality algorithms and requires new methods designed to exploit parallel computing; (ii) velocity requires real-time processing that introduces some uncertainties that new methods have to address; (iii) variety requires the availability of methods able to assess the quality of different data types and sources.

In order to address such issues we propose in this paper a Data Quality service able to provide information about the quality of the analyzed Big Data sources. Quality metadata are calculated and stored in order to let users, analytic or data mining applications be aware of the quality of input data and in particular to support the selection of relevant data and to identify "noises" that can affect data interpretation and/or decisions. In this paper we describe our experience in designing an architecture for the implementation of the DQ service. Mainly, we aim to show the main challenges we had to cope with. Note that we developed the service by considering real data gathered from a smart city case study in which analytic applications aim to offer advanced services to citizens and municipalities by analyzing public transportation data. The paper is structured as follows. Sect. 2 provides an overview of the overall architecture in which the DQ service is supposed to be implemented and integrated. Such architecture has been defined in the EUBra-BIGSEA project in which different services for Big Data are provided. Sect. 3 discusses the main challenges to address for assessing DQ in large data sets while Sect. 4 describes the DQ model that we adopt in this paper together with the different components that we implemented for assessing quality. Sect. 5 aims to present some issues that we have addressed during the implementation phase. Finally, Sect. 6 discusses previous contributions and highlights the novel aspects of the presented DQ service.

2 BACKGROUND AND MOTIVATIONS

The approach proposed in this paper has been defined by considering the scenario and related issues addressed by the EUBra project

IDEAS 2018, June 18–20, 2018, Villa San Giovanni, Italy

ACM ISBN 978-1-4503-6527-7/18/06…$15.00
https://doi.org/10.1145/3216122.3216124

called BIGSEA[1]. In this section we describe the project and the data sources used in the case study in order to better clarify the motivations behind the respective design choices.

The goal of the EUBra-BIGSEA project was to develop a cloud platform for Big Data management and exploitation. To this aim, cloud services able to empower Big Data analytics and thus able to support the development of data processing applications have been designed by considering Big Data issues and QoS and privacy and security constraints. In particular, the BIGSEA cloud services manage the following main operations: data ingestion, data search and filtering, data analytics and mining. Such services are provided in order to support users and/or applications in retrieving data stored in the platform and launching value-added analysis.

As regards the data, the project relied on data coming from several sources directly or indirectly related to the public transportation service of the Brazilian city of Curitiba.

For the design of the DQ service we focused on the streaming bus data. This source is composed of two logs containing monitoring information of complementary aspects of the public transportation service. The first log, *BusGPS*, contains information about the position of the vehicles. This position is retrieved through a GPS sensor located on each vehicle during its service time at an unknown sampling rate. Each line in the log contains: (i) the code assigned to the monitored vehicle, (ii) the longitude and latitude recorded by the GPS sensor, (iii) the timestamp at which the position has been recorded and (iv) the code assigned to the bus line to which the monitored vehicle belongs.

The second log, *BusUsers*, contains information about ticket validations of the users. The validation is executed using a magnetic card associated with a code. Each line in the log contains the following information: (i) the code assigned to the bus line in which the ticket validation has been performed, (ii) the name assigned to the bus line, (iii) the code assigned to the monitored vehicle, (iv) the code associated with the user magnetic card and (v) the timestamp at which the ticket validation has been recorded.

As stated above, the goal of the DQ evaluation is to provide quality metadata able to support data mining applications that, in this way, can have two different benefits:

- Awareness of the quality of the result as a consequence of the quality of the input data.
- Selection of a proper dataset, which satisfies DQ constraints.

3 DATA QUALITY ASSESSMENT ISSUES IN BIG DATA

As stated in Sect. 1, DQ is a prerequisite to get valuable results from analytic applications. In fact, DQ research area aims to develop methods and models for assessing and improving the quality level of the datasets and thus for judging data veracity and the suitability and usefulness of data for the context in which they are supposed to be used. Big Data complicate the achievement of such goal adding variety, volume and velocity issues. In this section we discuss the challenges that such issues raise.

[1]http://www.eubra-bigsea.eu/

3.1 Context-dependent Data Quality assessment

Variety refers to the fact that Big Data are gathered from heterogeneous data sources such as social networks, sensors or structured database. Many of such sources of Big Data are quite new [11]. Therefore, for assessing Big Data quality, it is not sufficient to consider the traditional data quality dimensions (e.g., accuracy, completeness, consistency and timeliness): additional dimensions need to be considered. For instance, the large number of sources makes trust, credibility and data provenance important to provide probability that data values are correct by considering data providers and all the operations/applications that manipulate data [8][1].

Considering the heterogeneity of sources, another aspect to consider is the fact that the relevance of the dimensions and related assessment algorithms change on the basis of the type of source and on the type of data. For example, completeness in structured data is evaluated by considering missing atomic values while assessing completeness in unstructured data requires considering the text at a different granularity level. Considering the data type instead, it is worth to note that the assessment of the correctness (i.e., accuracy) of a string needs an algorithm different from the one required for the assessment of the correctness of a number.

DQ assessment also depends on the users or applications that aim to access and use the data sources [1]. In fact, requirements can influence the selection of the set of DQ dimensions to consider and the metrics with which they have to be assessed. For example, assuming that timeliness is evaluated by considering the formula [5]:

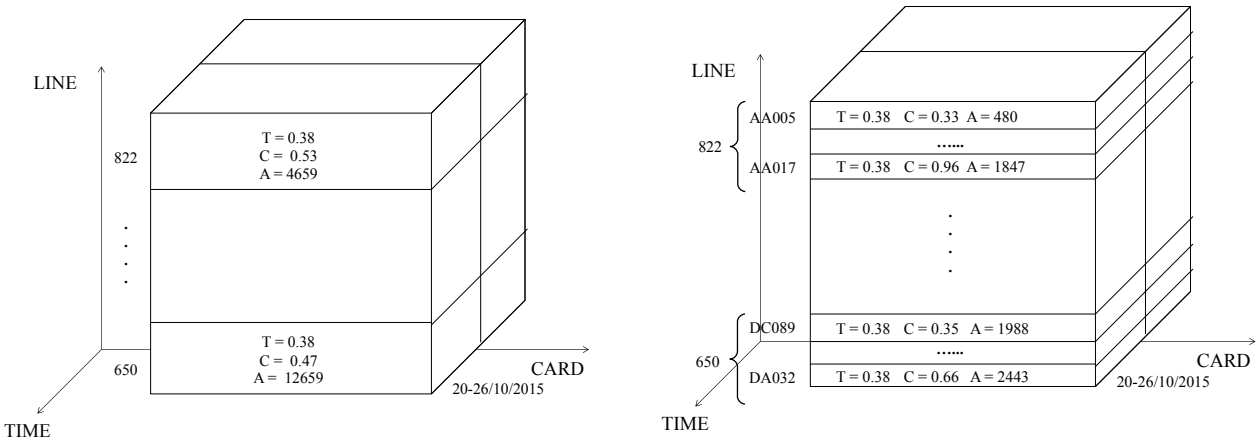
$$T = max\left(0, 1 - \frac{currency}{volatility}\right) \quad (1)$$

the need of an application for gathering current or historical values changes the data volatility (i.e., the temporal period in which data remain valid) and thus influences the evaluation of the entire timeliness dimension.

Defining the data types, data sources and applications as the *assessment context*, we can state that DQ evaluation cannot be performed in an objective way but we need to design an adaptive DQ service that, on the basis of such context, is able to select the right dimensions and assessment metrics.

3.2 Multi-granularity assessment

The selection of the dimensions provides the intensional perspective of the DQ metadata that have to be evaluated. The extensional view (i.e., the amount of values for each DQ dimension) is related to the granularity with which the assessment has to be performed. *Granularity* can be defined as the level of detail of data to which the DQ metrics should be evaluated. This is mainly related to users' or applications' requirements and thus to the usage of data. In fact, DQ computation at different granularity levels might be exploited by the users/applications for navigating Data Quality values and better understanding issues that affect the source reliability. Note that the same data source can have important variations in DQ evaluation at different granularity levels considered for the analysis. As an example, in Fig. 1 the comparison of the DQ at different granularity levels is represented using an hypercube for the *BusUsers* source. Here, the considered DQ metrics are timeliness (T), completeness



(a) Data quality values at the bus line granularity level

(b) Data quality values at the vehicle granularity level

Figure 1: Data quality navigation hypercube

(C), and amount of data (A). As can be observed, differences in DQ can be obtained for different attributes. In Fig. 1(a), it is shown that data related to two different bus lines can be characterized by an important difference in DQ values. The drill down operation, shown in Fig. 1(b), highlights how for the same bus line, Data Quality may change when focusing on the specific vehicles belonging to it.

The evaluation of Data Quality at different granularity levels raises challenges in designing the assessment algorithms. In fact, on the one hand effective aggregation methods have to be defined for unstructured sources and data streams; on the other hand, performance issues have to be considered for switching from fine to coarse granularity and viceversa.

3.3 Volume and time constraints management

The time of the analysis increases as the source becomes bigger, and so, if there are time constraints related to the execution of the DQ Assessment some approaches suggest to perform the analysis on a sample in order to obtain results as most similar as possible to the real ones in a limited time.

However, the analysis of a portion of a data source affects the reliability and correctness of the analysis and it is important that the users that access quality metadata are aware of this. To this aim, it is necessary to define an index able to express the quality of the metadata on the basis of the portion of data used in the assessment procedure. Beside the definition of such index there is also the problem of the selection of the type of sampling used to select the portion of data to evaluate. In fact, it may affect the computation of some Data Quality metrics. Literature contributions provide different ways to derive a sample (e.g., random sampling, sequential random sampling, stratified sampling), which may be selected and combined according to the features of the data source and of the quality dimensions of interest. The selection of the sampling method and the way in which the Data Quality metadata are evaluated is dependent on the quality metrics and the application requirements.

3.4 Data streams management

Big Data are characterized by velocity [11]: the speed of data creation makes Data Quality assessment challenging. In fact, data arrive continuously, making previous evaluations of Data Quality obsolete in a short time. To deal with this issue, data quality metrics on a data source should be updated in three cases:

- On demand: Data Quality assessment is performed since a request is submitted. Such request can be originated by the users/applications using the data source.
- Periodically: the Data Quality assessment can be performed at a fixed pace, according to the update frequency of the data stream. In our example, Data Quality values were updated on a daily base.
- Event-driven: some events in the execution environment may require an immediate update of the Data Quality values for a data source.

At each update, computation can start from the old Data Quality values, getting advantage of previous evaluations in order to reduce the computational time and cost. According to this, we consider DQ of a data source as a continuously refined value, which is computed incrementally starting from previous information. In order to enable the update process: (i) the incremental algorithms should be defined and (ii) some additional data needed for performing the update incrementally have to be kept, together with the historical Data Quality values. Keeping the computational cost and the storage space under control is not trivial.

4 A SOLUTION FOR ASSESSING DATA QUALITY IN BIG DATA

Considering the issues described in the previous section, we propose a solution that has been designed and developed within the BIGSEA project described in Sect. 2. Such solution is called *Data Quality Service* and provides an architecture and a methodology for assessing DQ in a Big Data scenario. Since the concept of DQ needs to be adapted to fit this new scenario, we, first of all, describe the

properties of DQ (Sect. 4.1). Then, we introduce the DQ assessment architecture and its components (Sect. 4.2).

4.1 Data quality model

A *Data Source* ds_i provides the data that have to be analyzed. It is characterized by a name and a description that defines the different data items (i.e., attributes) that the source includes. Each data item is characterized by a type (e.g., numeric, string, time, date). Each generic data source is also characterized by a set *DQ* of *Data Quality dimensions* that includes all the criteria dq_k that can be assessed for it. This set depends on the features of the data source and on its attributes. A DQ dimension is characterized by a name and a metric used to describe its quantitative value and it is assessed by an *Assessment Function* that is dependent on the type of data and the application requirements. In this way, the Data Quality assessment of a data source is defined on the basis of the context as defined in Sect. 3.1.

The assessment is performed on a *data object*. The data object is the portion of the data source ds_i that we consider for the quality evaluation: the data object can be either the entire source, a subset of it, or the result of a selective query limiting the data items or the values considered.

As described in Sect. 3.2 the assessment can be performed at different granularity levels. In fact, the result can be (i) an atomic value for each data item, (ii) an aggregated number that provides the quality level for an entire dataset or a portion thereof. The DQ service can assess the quality within four granularity levels:

- *Data source granularity level*: this level describes the DQ of the data source as a whole. An aggregated value for each quality dimension reflects the quality of the entire dataset.
- *Data object granularity level*: this level considers the Data Quality of the portion of data of interest for an application. It can be considered as the result of a query on the data source, which affects both the set of attributes and the number of entries to consider. As an example, an application interested in the *BusUsers* data source could select only data collected in a time interval for the analysis (number of entries reduction) and discard the information about the name assigned to the bus line (attribute reduction).
- *Attribute granularity level*: the assessment provides more detailed information about the Data Quality of specific attributes of the data source. Aggregated values of each dimension are provided for each attribute.
- *Value granularity level*: at this level the values of each attribute are used as grouping keys and then a quality value will be calculated and returned for each group. Thus the assessment is performed for each different value of each attribute (see for example the analysis for the bus lines depicted in Fig. 1(a)).

Note that the quality metadata are stored in a repository associated with the timestamp at which the assessment has been executed. The assessment function can gather values from such repository if, for example, it has to evaluate an aggregated quality level also considering historical values.

4.2 Data Quality service architecture

In this section, we describe the components of the Data Quality service. The proposed Data Quality Service Architecture is depicted in Fig. 2.

The main component of the architecture is the *DQ profiling and assessment* module. This module is in charge of assessing DQ dimensions of a data source. The module is composed of two main parts: the *DQ Profiling and Batch Assessment* module and the *DQ Context-aware Assessment* module.

The *DQ Profiling and Batch Assessment* module is in charge of evaluating general DQ features for the entire data source, which can be used as a reference by all the users and applications. It is executed for a first time when the data source is registered, and periodically in order to keep the information updated. The goal of this module is to provide a general profile and assessment of the data source. More specifically, the *profiling* extracts statistics and information about the data in the data source (e.g., types of values, number of missing values, number of distinct values) storing this information in the Quality metadata repository. It also performs an analysis to detect consistency rules among the attributes of the dataset. The *batch assessment* consists in providing an evaluation of DQ dimensions, which is not dependent from the context of the application. In this phase, a subset of the Data Quality dimensions are evaluated at the maximum granularity level for the whole Data Source. This preliminary evaluation is useful for all the applications willing to use the data, giving a general overview of the quality of the data source. Not all the quality dimensions can be evaluated for a given data source. The set of valid dimensions is automatically defined by the *DQ Profiling and Batch Assessment* when the source is registered by detecting the structure of the data source and the type of attributes provided and extracting the valid quality dimensions according to this information. The rules for associating a data type with the list of suitable data quality dimensions is contained in the *Quality metadata* repository. As stated before, this module is executed independently from the user requests, and it is re-executed periodically to update the pre-computed values, thus answering the data streams management issue described in Sect. 3.4.

To answer to the need of a context-dependent Data Quality evaluation, the user or application that aims to use the data source specifies its requirements using the *Requirements Specification* module. This module provides interfaces for helping the user in properly selecting a data object and specifying the set of relevant quality dimensions.To perform this task, the module accesses the profiling information of the data source contained in the *Quality metadata* repository, thus providing details on the data source attributes and values from which the user can indicate its selection, and on the valid Data Quality dimensions for the selected data object. The module also enables users to specify constraints on the desired values for each quality dimension. Starting from this information, the module automatically selects the subset of data that fits the requirements (e.g., select only data objects with completeness greater than 85%). All the users/applications settings are stored in the *Custom Settings* repository in order to build a configuration that is used to invoke the Data Quality service and to execute the assessment. Preferences are also saved for the subsequent invocations.

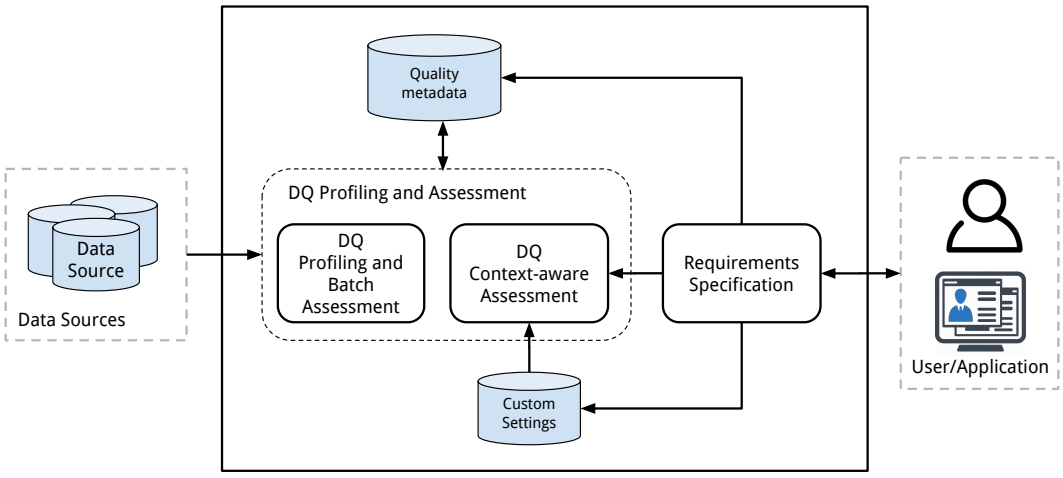


Figure 2: Data Quality Service Architecture

The *DQ Context-aware Assessment* uses the requirements expressed through the *Requirements Specification* module as an input for providing a context-aware evaluation of the Data Quality. The context-aware assessment phase performs the assessment considering different granularity levels (i.e., data source, data selection, attribute and value granularity level), answering to the multi-granularity assessment issue discussed in Sect. 3.2. Despite the assessment performed by the *DQ Context-aware Assessment* module is limited to a subset of the data source specified in the data object, the context-aware assessment might require a lot of computation time according to the size of the data object, the granularity levels required, and the number of dimensions to be considered. Since responsiveness in this phase of explorations of the data source properties might be important for the users, the *DQ Context-aware Assessment* module contains a *DQ Adapter* that tunes the precision of the results according to the specification of the user. If fast responses are needed, the adapter can reduce the response time by selecting a subset of the data object, providing a faster evaluation but with a lower precision. To provide a measurement of the reliability of the results, dependent on the portion of the data object analyzed in place of the whole portion required in the user request, we defined an index called *Confidence*. The DQ adapter is our proposal for solving volume and velocity issues described in Sect. 3.3.

According to what we have described in this section, the two modules composing the *DQ Profiling and Assessment* enable both a periodic off-line objective evaluation and an on-line and users dependent evaluation. The off-line evaluation is very relevant when dealing with Big Data because it limits the on line analysis only to the particular requests, thus reducing the response time.

The output of the *DQ Profiling and Assessment* module is a set of metadata expressing a Data Quality evaluation of the sources, coupled with a precision value. This information is stored in the *Quality Metadata* database.

5 IMPLEMENTATION DETAILS AND LESSONS LEARNED

The DQ service described in Sect. 4 has been designed and tested by considering the data sources (i.e., BusGPS and BusUsers) described in Sect. 2. In this section we describe the details of the implementation, initial findings, and lessons learned.

5.1 Data Quality Profiling and Assessment module

The *DQ Profiling and Batch Assessment* module in our implementation provides, for all the attributes contained in the data source, the following information: number of values, number of null values, number of distinct values, maximum, minimum, mean, and standard deviation (only for numerical values). It is also able to evaluate the following DQ dimensions:

- *Accuracy*: it is defined as the degree with which a value is correct [13]. Currently, we have implemented it only for numerical values, in order to check if they are included in an expected interval or they are outliers (and thus not accurate).
- *Completeness*: it measures the degree with which a dataset is complete [13]. It is evaluated by assessing the ratio between the amount of values currently available in the dataset and the expected amount of values. The expected amount of values considers both null values in available registrations and missing registrations. Note that, as regards data streams, missing registrations are easy to detect if data are sensed with a specific frequency. If data are not collected at a regular pace it is possible to rely on historical data to estimate the sampling frequency that often varies over time.
- *Consistency*: it refers to the violation of semantic rules defined over a set of data items [2]. Therefore, this dimension can be calculated only if a set of rules that represent dependencies between attributes is available. We have developed a module that detects functional dependencies and checks if the values in the dataset respect them.

- *Distinctness*: it is related to the absence of duplicates and measures the percentage of unique registrations or distinct attribute values in a dataset.
- *Precision*: this dimension can be calculated only for numerical attributes and can be defined as the degree with which the values of an attribute are close to each other. In particular, precision is derived by considering the mean and the standard deviation of all the values of the considered attribute.
- *Timeliness*: it is the degree with which values are temporally valid. We evaluate it by considering the formula already described in Eq. 1.
- *Volume* (or Amount of Data): this quality dimension provides the number of values contained in the analyzed data source.

As explained in the Sect. 4.2, the context-aware assessment is based on the requirements specified through the *Requirement Specification* module. Requirements might contain all the following data: source to analyze, output folder in which the assessment results have to be stored, list of the attributes to consider in the analysis, filters to select the values to include in the Data Object to analyze, Data Quality dimensions to evaluate, granularity, Data Quality requirements, additional consistency rules.

Once that the *DQ Context-aware Assessment* module receives this information, it retrieves the source, identifies the Data Object by applying the desired selection and projection operations and starts to assess the quality metadata computing the requested dimensions with the desired granularity (or retrieving them from the *Quality metadata* repository if already available).

5.2 Execution environment

In order to develop and test the modules, an Apache Spark environment has been considered over a Hadoop Distributed File System - HDFS. This allows the distribution of the work among multiple machines over a parallel collection of data in the form of Resilient Distributed Datasets (RDD). In this way data are organized in multiple partitions and using the fast RAM memory of each machine the analysis time is considerably reduced and Big Data sources can be analyzed. In details, parallel programming is achieved by dividing the operations on the RDD in multiple smaller operations that can be assigned to each machine and by aggregating the results when all the machines finish their operations. The writing of these parallel algorithms is eased by the availability of a lot of APIs, especially in Python and Scala, that contain multiple simple functions that allow developers to command very complex operations. We have developed the Data Quality assessment service functionalities in Python.

From the point of view of the implementation, the distributed environment offers some pros but also comes with some cons. In fact, on the one hand, distributed approaches are suitable to perform parallel processing and thus reduces the execution time and increases flexibility. On the other hand the manageability of the whole system is more difficult. Indeed, each machine has to be appropriately configured: it has to access data and, if special libraries are needed, they have to be installed on all the working nodes.

In details, for the development, we use the Hortonworks SandBox In Virtual Machine TM with i7 6700HQ Processor (4 physical cores @2.60GHz, 8 thread), 12Gb of DDR3L 1600MHz RAM and SSD disk containing the data (Write/Read). The Hortonworks Sandbox is a portable Apache Hadoop environment that contains several services among which there are HDFS, Apache Spark and Apache Zeppelin that is a web-based notebook which brings data exploration, visualization, sharing and collaboration features to Spark. For executing the algorithms in a cloud environment we rely on Azure Cluster, with Xeon processors, from 8 cores up to 48 cores @3GHz and from 12Gb up to 52Gb of RAM.

5.3 Development issues

During the implementation of the algorithms we had to solve some issues and consequently change the code and design choices. We had to address problems both in the loading and processing stage. In the loading stage, it is necessary to consider that encoding issues can occur and some pre-loading transformation actions need to be performed. Furthermore, for improving the flexibility of the platform to manage heterogeneous sources we decided to design the Source Analyzer that is in charge of the registration of the sources and of performing a preliminary analysis to automatically identify the type of the data items stored in the source. In our case study, such procedure failed for some items that were considered numbers but they could not be treated as numbers. For example, the numbers of the users' cards and the GPS coordinates are classified as numbers but aggregation operations such as the average, max, min or the evaluation of the precision have no meaning. Therefore, this operation cannot be completely automatic but the system needs additional knowledge on the domain. In the processing stage, as described in Sect. 3.4 we dealt with data streams and we analyzed their quality by applying a blocking technique. Thus, we evaluated the profiling metrics and the Data Quality dimensions for each block and then, on the basis of the dimensions, we aggregated the computed values to obtain the assessment associated with the entire source (composed of blocks analyzed in the past). For each dimension, we had to define suitable aggregation methods since for most of the dimensions it is not possible to aggregate the new and old quality values with a simple function as the average.

Once we fixed all the problems, we started to run the algorithms considering the BusGPS and BusUsers sources.

5.4 Experiments results and findings

For the BusUsers, we considered the data collected between October 2015 and September 2016, in which a total of 11 Gigabyte of data, analyzed at day-frequency, have been correctly analyzed and in which 3.1 Mb of data were not assessable due to a different structure of the data or to the presence of corrupted attributes and records. To perform this test the Azure cluster was used with 1 worker (or datanode), 4 executor nodes, each one with 2 cores and 2 Gigabyte of RAM, and the master node of the Spark application with 4 Gigabyte of RAM.

In order to better clarify how the system works, we want to add some details of the analysis that has been performed on this source. First of all, the first time that the source has been uploaded into the system it has been analyzed in order to automatically detect the type of the collected attributes. The results of this task are reported in Tab. 1.

Attribute	Type
CODLINHA	string
CODVEICULO	string
DATAUTILIZACAO	%d/%m/%y %H:%M:%S
NOMELINHA	string
NUMEROCARTAO	float

Table 1: Source description

As described in Sect. 3.1, the Data Quality assessment depends on the type of the attributes. For this reason, exploiting experts knowledge coded in a specific repository, on the basis of their type, the *DQ Profiling and Batch Assessment* module associates the attributes with the quality dimensions that can be evaluated and the related granularity level (see Tab. 2).

Note that as discussed in Sect. 5.3, the number of users' cards was considered as a number and then the system wrongly associates accuracy and precision with this attribute. Here, a domain expert is needed to check and correct the results.

Considering the source level granularity, Tab. 3 describes the results obtained from the *DQ Context-Aware Assessment* of the BusCard dataset at the source level.

As regards completeness, there were not missing values in the dataset but there were some missing registrations. These registrations have been detected by considering historical values and estimating the frequency with which users were expected to enter the buses. Consistency has been measured by considering the functional dependency between the CODLINHA (code of the bus line) and NOMELINHA (name of bus line). The Distinctness value reveals that no duplicated rows were included in the dataset. The Timeliness value is the mean value of the timeliness associated with the considered registrations. The total number of registrations is shown by the volume dimension.

Just to provide an idea of the execution time, the assessment of 26 dimensions on the entire set of historical data and for all the possible granularities took 6 hours, 10 minutes and 38 seconds, and was obtained by considering also the overheads regarding the initialization of the executors. The analysis of most of the dimensions took less than 15 seconds for each data block (containing the streaming data of one day).

The execution time resulted to be higher for the BusGPS source. This is due to the fact that such source is larger: the amount of data collected between October 2015 and September 2016 is 43 Gigabyte. The analysis of quality dimensions on each block (i.e., data referred to one day) took in average 100 seconds. This shows the natural correlation between volume and execution time.

We tried to decrease the execution time by increasing the number of computational nodes. In general, this can work but we found out that for the elasticity property of a cloud environment it is difficult to estimate the execution time. In fact, it is necessary to consider that machines can interfere and computation resources have to be shared with other applications. In this case even if we increase the number of nodes, the reduction of execution time cannot be given for granted. In fact, it might happen that we do not have the exclusive use of the resources and thus the execution time can also increase.

6 STATE OF THE ART

The importance of addressing veracity in Big Data and thus of properly evaluating and managing the quality of data has been widely discussed in the literature. For example, [1] claims the importance, in the Big Data scenario, of redefining the DQ dimensions on the basis of data type, sources and applications considered.

Big Data Quality dimensions have been also analyzed in [6]. The assessment process here depends on the goals of data collection and thus also on the considered business environment and the involved data sources.

Another contribution, [12], defines the concept of quality-in-use of Big Data. Authors define the concept of Adequacy of data as "the state or ability of data of being good enough to fulfill the goals and purposes of the analysis". In particular, Adequacy is considered as composed of three aspects: the degree with which the datasets can be used in the domain of interest, the temporal validity of data for the specific analysis and the quality of the resources available to process data. These three concepts are introduced to reclassify the Data Quality dimensions defined in the ISO/IEC 25012 standard. In short, the authors propose a general assessment architecture but they provide a general overview without discussing details and issues.

Other papers focus on the novel dimensions that should be introduced in the Big Data scenario. Authors in [9] discuss the rise of Big Data on cloud computing and consider data consistency as the most important dimension in this field: the quality of different data sources is high if there are not inconsistencies among their values. [4] highlights instead the importance of the trustworthiness. Trust is also considered in [7]: this paper focuses on data mining systems and claims that data validation and provenance tracing become more than a necessary step for analytics applications.

All these papers confirm the motivations behind our work: Data Quality dimensions definition and assessment algorithm have to be revisited and are strongly dependent on the type of data, data source and the application that requests data. In this work, we propose an architecture for an adaptive Data Quality service able to provide the right quality metadata for the considered application. It is important also to highlight that the assessment we propose allow users to gradually explore the quality of the source: they can move from a general to a more detailed analysis, with a variable level of details.

The presented Data Quality service has been implemented and it is able to manage different types of sources. Some approaches presented in the literature are very specific for solving a Data Quality issue. [14] focuses on sensor networks and proposes a method calculate different quality indicators and aggregate them on different time scales. The quality indicators are calculated using a sliding window model and thus considering only the k elements within the window. [10] considers the problem of entity resolution and proposes a MapReduce version of the sorted neighborhood blocking algorithm. In [3] authors consider the veracity issue in Big Data and describe different methods for detecting the true values

DQ dimension	Data Source/Object	Attribute	Value
Accuracy	-	Yes, NUMEROCARTAO	-
Completeness	Yes	Yes, All attributes	Yes
Consistency	Yes	-	Yes
Distinctness	Yes	Yes, All attributes	Yes
Precision	-	Yes, NUMEROCARTAO	-
Timeliness	Yes	Yes, All attributes	Yes
Volume	Yes	Yes, All attributes	Yes

Table 2: Data Quality dimensions that can be assessed on the considered source

DQ dimension	Value
Completeness	0.99
Consistency	0.99
Distinctness	1
Timeliness	0.61
Volume	75565968

Table 3: Source quality

within a dataset. In this paper we want to provide a more general and adaptive solution for Data Quality awareness, supporting users and application in the evaluation of the reliability of their analysis.

7 CONCLUSIONS

In this paper, we have discussed the issue of providing Data Quality awareness for a successful exploitation of Big Data. The quality of the data in input has always been a key factor for ensuring the quality of the results provided by analytic applications. In a Big Data era, the assessment of Data Quality is even more complex than in the past, since we have to deal with unstructured, continuously updating, and multi-source generated information.

In the framework of the EUBra-BIGSEA project, we addressed these issues by designing a Data Quality assessment architecture. In our approach, we had to face several issues both at design time (for managing streaming data in a context-aware manner at different levels of granularity) and in the implementation phase (managing distributed computation and interferences among applications). Note that the proposed approach is still under refinement. On the one hand more sources have to be considered in order to test the Data Quality service. On the other hand, the quality assessment works by considering one source at the time while many applications analyze integrated sources. Future work will also focus on the quality assessment of a dataset resulting from a multi-source integration.

ACKNOWLEDGMENT

The authors would like to thank the Curitiba City Hall for the data sources. The authors work has been partially funded by the EUBra-BIGSEA project by the European Commission under the Cooperation Programme (MCTI/RNP 3rd Coordinated Call), Horizon 2020 grant agreement 690116.

REFERENCES

[1] Carlo Batini, Anisa Rula, Monica Scannapieco, and Gianluigi Viscusi. 2015. From Data Quality to Big Data Quality. *J. Database Manag.* 26, 1 (2015), 60–82. https://doi.org/10.4018/JDM.2015010103

[2] Carlo Batini and Monica Scannapieco. 2016. *Data and Information Quality - Dimensions, Principles and Techniques.* Springer. https://doi.org/10.1007/978-3-319-24106-7

[3] Laure Berti-Equille and Javier Borge-Holthoefer. 2015. *Veracity of Data: From Truth Discovery Computation Algorithms to Models of Misinformation Dynamics.* Morgan & Claypool Publishers. https://doi.org/10.2200/S00676ED1V01Y201509DTM042

[4] Elisa Bertino. 2015. *Data Trustworthiness—Approaches and Research Challenges.* Springer International Publishing, Cham, 17–25.

[5] M. Bovee, R.P. Srivastava, and B.R. Mak. September 2001. A Conceptual Framework and Belief-Function Approach to Assessing Overall Information Quality. In *Proceedings of the 6th International Conference on Information Quality.* Boston, MA.

[6] Li Cai and Yangyong Zhu. 2016. The Challenges of Data Quality and Data Quality Assessment in the Big Data Era. *Data Science Journal* 14 (2016), 2.

[7] Dunren Che, Mejdl Safran, and Zhiyong Peng. 2013. *From Big Data to Big Data Mining: Challenges, Issues, and Opportunities.* Springer Berlin Heidelberg, Berlin, Heidelberg, 1–15.

[8] Chenyun Dai, Dan Lin, Elisa Bertino, and Murat Kantarcioglu. 2008. An Approach to Evaluate Data Trustworthiness Based on Data Provenance. In *Secure Data Management, 5th VLDB Workshop, SDM 2008, Auckland, New Zealand, August 24, 2008, Proceedings.* 82–98. https://doi.org/10.1007/978-3-540-85259-9_6

[9] Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, and Samee Ullah Khan. 2015. The rise of "big data" on cloud computing: Review and open research issues. *Inf. Syst.* 47 (2015), 98–115. https://doi.org/10.1016/j.is.2014.07.006

[10] Lars Kolb, Andreas Thor, and Erhard Rahm. 2012. Multi-pass sorted neighborhood blocking with MapReduce. *Computer Science - R&D* 27, 1 (2012), 45–63. https://doi.org/10.1007/s00450-011-0177-x

[11] Andrew McAfee and Erik Brynjolfsson. 2012. Big Data: The Management Revolution. *Harvard Business Review* 90, 10 (2012), 60–68.

[12] Jorge Merino, Ismael Caballero, Bibiano Rivas, Manuel A. Serrano, and Mario Piattini. 2016. A Data Quality in Use model for Big Data. *Future Generation Comp. Syst.* 63 (2016), 123–130. https://doi.org/10.1016/j.future.2015.11.024

[13] Richard Y. Wang and Diane M. Strong. 1996. Beyond Accuracy: What Data Quality Means to Data Consumers. *J. of Management Information Systems* 12, 4 (1996), 5–33. http://www.jmis-web.org/articles/1002

[14] Wenlu Yang, A. Da Silva, and M. L. Picard. 2015. Computing data quality indicators on Big Data streams using a CEP. In *Computational Intelligence for Multimedia Understanding (IWCIM), 2015 International Workshop on.* 1–5. https://doi.org/10.1109/IWCIM.2015.7347061