Analyzing the Quality of Twitter Data Streams

Franco Arolfo[1] · Kevin Cortés Rodriguez[1] · Alejandro Vaisman[1]

Abstract

There is a general belief that the quality of Twitter data streams is generally low and unpredictable, making, in some way, unreliable to take decisions based on such data. The work presented here addresses this problem from a Data Quality (DQ) perspective, adapting the traditional methods used in relational databases, based on quality dimensions and metrics, to capture the characteristics of Twitter data streams in particular, and of Big Data in a more general sense. Therefore, as a first contribution, this paper re-defines the classic DQ dimensions and metrics for the scenario under study. Second, the paper introduces a software tool that allows capturing Twitter data streams in real time, computing their DQ and displaying the results through a wide variety of graphics. As a third contribution of this paper, using the aforementioned machinery, a thorough analysis of the DQ of Twitter streams is performed, based on four dimensions: Readability, Completeness, Usefulness, and Trustworthiness. These dimensions are studied for several different cases, namely unfiltered data streams, data streams filtered using a collection of keywords, and classifying tweets referring to different topics, studying the DQ for each topic. Further, although it is well known that the number of geolocalized tweets is very low, the paper studies the DQ of tweets with respect to the place from where they are posted. Last but not least, the tool allows changing the weights of each quality dimension considered in the computation of the overall data quality of a tweet. This allows defining weights that fit different analysis contexts and/or different user profiles. Interestingly, this study reveals that the quality of Twitter streams is higher than what would have been expected.

Keywords Data quality · Social networks · Twitter · Big data

1 Introduction and Motivation

The relevance of Big Data has been acknowledged by researchers and practitioners even before the concept became widely popular through media coverage (The Economist. Data 2008). Although there is no precise and formal definition, it is accepted that Big Data refers to huge volumes of heterogeneous data ingested at a speed that cannot be handled by traditional database systems tools, and characterized by the well-known "4 V's" (volume, variety, velocity, and veracity). That means, not only the data volume is relevant, but also the different kinds of structured, semi-structured and unstructured data, the speed at which data arrives (e.g., real time, near real time), and the reliability and usefulness of such data. However, it is also acknowledged that most of the promises and potential of Big Data are far from being realized at the moment of writing this paper. This gap between promise and reality is due to the many technical problems and challenges that are usually overlooked, although the database research community has warned about them. These problems refer, among many ones, to heterogeneity, scale, timeliness, complexity, and privacy. Moreover, in Agrawal D. et al. (2011), the whole Big Data process is split in five phases, and it is stated that from the data acquisition phase to the results interpretation phase, data quality (DQ) plays a key role. The classic concepts of DQ used in relational databases must be revisited in a Big Data context, since many new problems appear, which are not present in traditional relational database scenarios (Saha and Srivastava 2014; Cai and Zhu

✉ Alejandro Vaisman
avaisman@itba.edu.ar

Franco Arolfo
farolfo@itba.edu.ar

Kevin Cortés Rodriguez
kcortesrodrigue@itba.edu.ar

[1] Department of Information Engineering, Instituto Tecnológico de Buenos Aires Lavardén 315, C1437FBG, Ciudad Autónoma de Buenos Aires, Argentina

2015; Firmani et al. 2015). Intuitively, each of the "V's" define a different context for data analysis, and therefore, for DQ. Thus, there is a strong relationship between the work about contexts in DQ (e.g., Ciaccia and Torlone (2011) and Poeppelmann and Schultewolter (2012)) and the problems of DQ in Big Data, since different notions of quality must be used for different types of data.

Twitter is a well-known microblogging service where users post messages, denoted "tweets". These are short messages with a maximum length of (currently) 280 characters. Often, hashtags are used for grouping and searching tweets according to some topic or subject. Since people who post those messages share preferences and opinions with other users, tweets are a valuable source of people's opinions and sentiments. Such tweets can thus be used to help in marketing tasks, social analysis, and so on. Such data can be analyzed online, to help taking immediate decisions, or can be cleaned, structured, and stored in a data warehouse or a data lake for historical analysis. It follows immediately that Twitter feeds conform a typical real-time Big Data scenario: data come at high speed, are highly unstructured, and, in principle, have very volatile reliability and usefulness (confirming or not these assumptions is one of the goals of this paper). All of these characteristics are the complete opposite of a relational database analytics scenario, where data are highly structured, and cleaned, transformed and analyzed offline. To be used in a decision-making scenario, Twitter feeds should have at least a minimum quality, to avoid erroneous decisions. For example, Soto et al. (2018) use a DQ approach to filter Twitter users in order to analyze tweets for policy making in health care, in order to obtain data useful for their needs. It follows that the quality of the data must be assessed accounting for the issues mentioned above. Further, it would be desirable to be able to give context to DQ, in the sense that Twitter data may be used for different goals and in different situations, which means that data quality dimensions should be addressed differently in each case.

Despite the relevance of the topic, there has been not much work so far, in particular regarding the implementation of quality processes over Big Data sources. For Twitter data, in particular, most of the work has focused on mining tweets and, mainly, on performing sentiment analysis on them, to discover the user's feeling behind a tweet. This analysis, although useful, is clearly incomplete for decision-making, because the former does not account for the quality of the data. A more comprehensive analysis from a DQ point of view is required, to make reliable and informed decisions based on Twitter data. Also, DQ can be used to determine the attributes that characterize the different quality of the tweets, filter out bad quality data, and/or validate the conclusions drawn in the data analysis phase. This paper tackles these issues, analyzing the quality of Twitter data feeds using DQ concepts adapted to this scenario. For this, it uses a software tool developed to capture Twitter streams in real time, and computing and displaying their DQ features.

1.1 Contributions

The contributions of this work are the following:

1. The definition of DQ dimensions and metrics in a Big Data scenario where data arrive as unstructured documents and in real time. Traditional quality dimensions are redefined, to address such characteristics. This general scenario is instantiated to study the concrete case of Twitter feeds.
2. A system that acquires tweets in real time, computes the quality of each tweet, applies the quality metrics defined formally in the paper, and displays a collection of graphics for analyzing the overall quality of the stream. The implementation allows: (a) filtering tweets using a set of keywords; (b) computing the overall data quality of a Twitter stream; (c) displaying each DQ dimension separately; (d) defining topics and computing the DQ of the tweets corresponding to each topic (e.g., politics, sports, arts); (e) displaying the DQ over a map, thus classifying the quality according with the geolocalization of the tweet; (f) Storing searches in an ElasticSearch database, in order to efficiently retrieve those searches when they are needed, for comparing results. In addition, the system accounts for the context in which tweets are analyzed, allowing to dynamically change the weights of the quality dimension metrics.
3. A thorough study of Twitter data streams in real time, using the machinery described above. Several tests are planned, executed, analyzed, and reported, to investigate the DQ of Twitter feeds from different dimensions. Surprisingly, as commented above, this analysis suggests that the quality of Twitter data streams is, in general, higher than the one most people would expect. Therefore Tweeter data can be considered very valuable for decision making. Moreover, the study shows that quality can vary for tweets corresponding to different topics, which also helps to assess the value of these data when they are used in particular contexts.

1.2 Paper organization

The remainder of the paper is structured as follows. Section 2 discusses related work. In Section 3, the traditional DQ dimensions and metrics are presented. Section 4 studies the DQ dimensions and metrics for Big Data and presents the computation of such metrics to evaluate the quality of a tweet. Section 5 describes the implementation of the system. Section 6 is the core of the paper, and presents the experimentation over Twitter data, and reports and discusses the results of this study. Section 7 concludes the paper.

2 Related Work

Ensuring data quality in relational databases has long been acknowledged as a relevant research topic in the database community. This research resulted in the definition of dimensions, metrics, and methods to assess the quality of a database (Wang and Strong 1996), and also in an ISO standard specification[1]. Despite this, classic research considers DQ as a concept independent of the context in which data are produced and used. This assumption is clearly not enough to solve complex problems, particularly in current times, when, among other facts, ubiquitous computing requires accounting for the space and time associated with a query. Strong et al. (1997) realized this problem, and claimed that the quality of data is highly dependent on the context, which became an accepted fact thereon. The rationale for this conclusion was based on the fact that, similarly to quality in general, DQ cannot be assessed independently of the consumers who choose and use certain products. As an example of these concepts, Wagner et al. (2011) proposes a system where contextual information allows evaluating the quality of blood pressure data. Further, with a more general perspective, Poeppelmann and Schultewolter (2012) introduces a framework that allows context-sensitive assessment of DQ, through the selection of dimensions for each particular decision-maker context and her information requirements.

It is widely accepted that most modern applications, particularly over the web, are required to be context-aware. Thus, there is a large corpus of work on the topic. Bolchini et al. (2007) presented a survey of context models, with a well-defined structure, that identifies some important aspects of such models. In particular, the work remarks that models must account for *space*, *time*, *context history*, *subject*, and *user's profile*. Preferences in databases have also been extensively studied (Ciaccia and Torlone 2011; Stefanidis et al. 2011). In the multidimensional databases domain, (Marotta and Vaisman 2016) proposes to define the context through the use of logic rules, representing the database as a first-order logic theory. The particularities of data quality in the context of Big data are addressed in Firmani et al. (2015) and Batini et al. (2015), which study how the "4 V's" mentioned in Section 1 impact on well-known DQ dimensions and metrics used in traditional structured databases (Batini and Scannapieco 2006). The main message in Firmani et al. (2015) is that Big Data quality should be defined in source-specific terms and according to the dimension(s) under investigation. In some sense, this means that the context is again present in a Big Data scenario when quality is addressed.

Social networks and, especially, Twitter, are increasingly becoming a source of data for information systems of many different kinds. As such, assessing the quality of data obtained from these sources is crucial for the success of information systems that rely on such data. Several works in various fields reflect this influence. In health sciences, for example, Zadeh et al. (2019) study the characteristics of Twitter data that were used to monitor flu outbreaks in the US, confirming that flu-related traffic on social media is closely related with actual flu outbreaks. In disaster management, Abedin and Babar (2018) argue that social media plays a significant role in the rapid propagation of information when disasters occur. The authors focus on the use of social media during the response phase, studying the use of Twitter by Emergency Response Organisations (EROs) during a fire hazard occurred in Victoria, Australia in February, 2014. Citizen science, where people contribute information for scientific research, has also been characterized as as an information quality research frontier (Lukyanenko et al. 2020). The work analyzes the quality of information created by ordinary people, which is known as user-generated content, through social networks like Facebook, Twitter, Instagram, or YouTube, online reviews like TripAdvisor, IMDB, and so on. Ye et al. (2019) studies how mobile data services (MDS) providers can leverage reviews posted in social media to innovate and profit from them, reporting that online reviews positively impact MDS popularity directly. Along similar lines, Chang and Chen (2019) propose a model based on sentiment analysis and credibility, to study the impact of online reviews (taking TripAvisor as a source for their experiments), reporting that negative emotions and low-credibility reviews have a high influence on hotel ranking. This study also reports that credibility has a higher influence than sentiment analysis on hotel ranking.

As a result of the above, it is clear that assessing the quality of data generated by social media users is a relevant research and practical problem, which has not been properly addressed yet. In this sense, regarding the analysis of the quality of Twitter data, most of the research work has focused on data mining tasks (mainly classification and text mining (Byrd et al. 2016)), and sentiment analysis (Hao et al. 2011; Fornacciari et al. 2015; Guruprasad et al. 2015). However, it seems that simply applying well-known techniques over these kinds of data misses an important point, namely the quality of the data over analysis.

[1] http://iso25000.com/index.php/en/iso-25000-standards/iso-25012

Recently, Soto et al. (2018) partially addressed the issue of DQ in some way, in the context of health care data analysis. The authors point out the problems generated by unreliable Twitter data for the analysis of such data, and also remark that DQ is overlooked in social media analysis. The paper analyzes people's opinions on different topics and from a certain region, accounting for DQ. However, the paper does not dive into DQ dimensions, and limits to provide an offline cleaning methodology. Also, Salvatore et al. (2020) proposed a framework for studying Twitter DQ, and a collection of good practices and indicator for such task. However, the study is limited to the case of the 2018 London marathon, reducing the scope. On the contrary, the study presented in this paper is performed on real-time and historical data obtained from the tool described here.

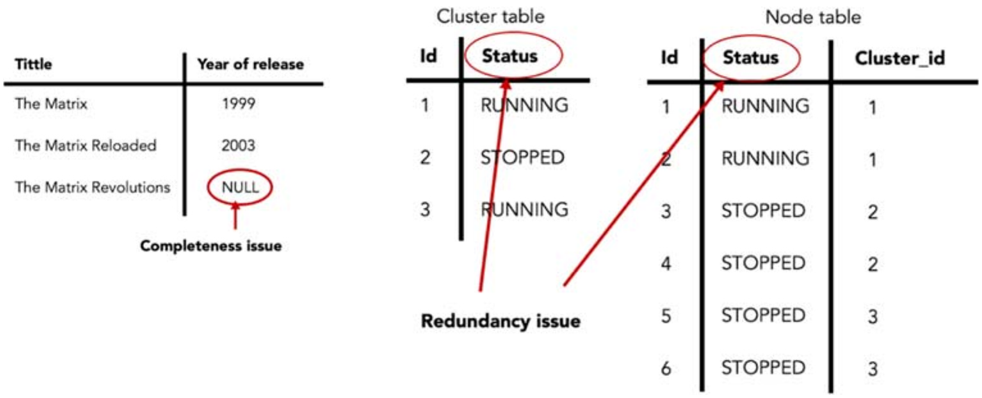
A preliminary version of the system discussed in this paper, was presented by Arolfo and Vaisman (2018) in previous work, which studies how data from Twitter can be captured, and their DQ can be computed in real time. The system presented here has a totally new and more robust and efficient architecture, as explained later in the paper, as well as an expanded and improved graphic machinery. Also, the system allows storing and retrieving past searches, and provides the capability of adapting the quality metrics weights to the user interests, accounting for context analysis.

3 Background on Data Quality

Data Quality (DQ) is a multi-faceted concept, represented by different dimensions, each one referring to a different quality aspect (Strong et al. 1997; Batini and Scannapieco 2006). A *dimension* captures a facet of DQ, while a *metric* is a quantifiable instrument that defines the way in which a dimension is measured. Since a DQ dimension is in general a wide concept, an associated metric allows specifying a concrete meaning for the dimension. As a consequence, many different metrics can be associated with the same DQ dimension, and their application will measure several different aspects of such dimension. In a broader sense, the *quality of an object or service* indicates to what extent this object or service fits the needs to solve a given problem. That is, quality is not absolute to the object or service *per se*, but relative to the problem to be solved. This is the approach followed in this work.

While a large number of DQ dimensions were proposed in the literature, there is a basic set of them, which are generally acknowledged to be representative of the quality of data (Batini and Scannapieco 2006; Scannapieco and Catarci 2002). This set includes accuracy, completeness, consistency, freshness (or timeliness), among other ones. These are described next, to make the paper self-contained.

– *Accuracy:* Specifies how accurate data are, and involves the concepts of *semantic accuracy* and and *syntactic accuracy*. The former refers to how close is a real-world value to its representation in the database. The latter indicates if a value belongs to a valid domain. In other words, accuracy describes the closeness between a value v and a value v', considered as the correct representation of the real-life phenomenon that v aims at representing. For example, if someone wants to type the name "John" but typed "Jhn", there is an accuracy issue.
– *Completeness:* Represents the extent to which data suffice for the task at hand. For relational databases, this can be characterized as the presence/absence and meaning of null values, assuming that the schema is complete. For example, the left-hand side of Fig. 1 displays the chronological order of the Matrix movies, although it can be seen that "The Matrix Revolutions" lacks the year of release value, therefore there is a completeness issue there. If a query asks for the directors of these movies, there is also a *schema completeness* issue.
– *Redundancy:* Refers to the representation an aspect of the world with the minimal use of information resources. For example, in Fig. 1 (center and right),



Tittle	Year of release
The Matrix	1999
The Matrix Reloaded	2003
The Matrix Revolutions	NULL

Completeness issue

Cluster table

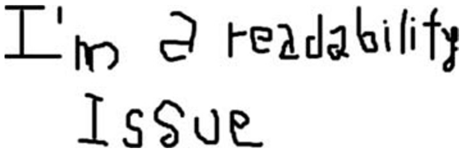
Id	Status
1	RUNNING
2	STOPPED
3	RUNNING

Node table

Id	Status	Cluster_id
1	RUNNING	1
2	RUNNING	1
3	STOPPED	2
4	STOPPED	2
5	STOPPED	3
6	STOPPED	3

Redundancy issue

Fig. 1 **a** Completeness issue (left); **b** Redundancy issue (center & right)



I'm a readability Issue

Fig. 2 Readability issue example

the nodes that compose the different clusters in an architecture are shown, together with their status. It can be seen that Cluster 3 has a "RUNNING" status, but its nodes are "STOPPED". Redundancy here also caused an *inconsistency* issue (see below).

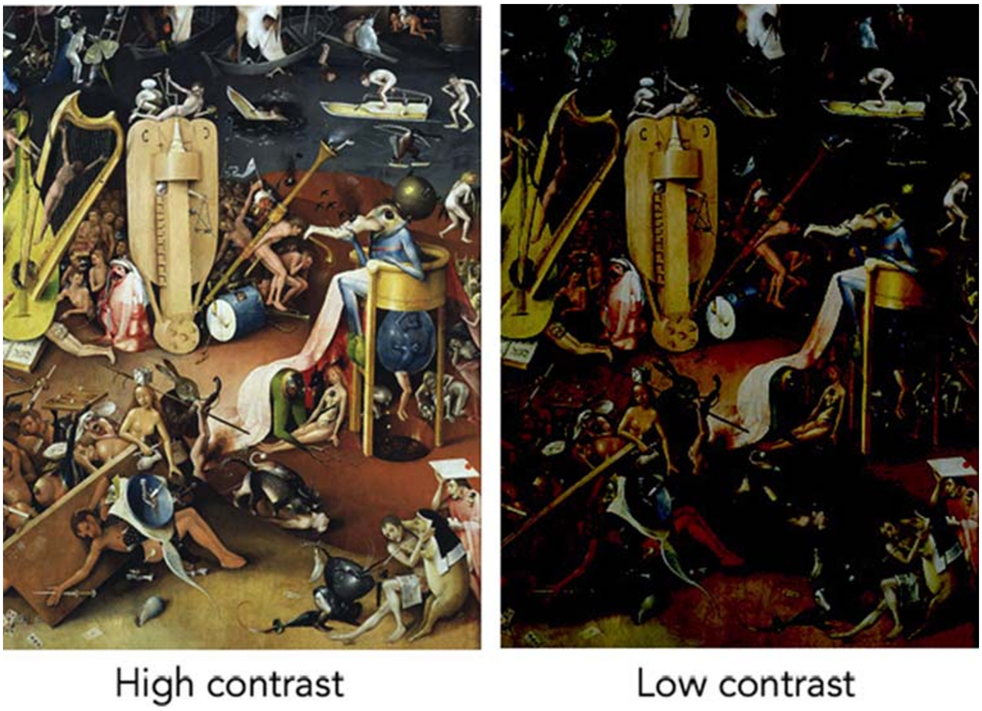
- *Consistency:* Refers to the capability of the information to comply without contradictions with all the rules defined in a system. For example, in a relational database constraints are defined to guarantee consistency. As commented above, Fig. 1 (right) shows a *consistency* issue.
- *Readability:* Refers to the ease of understanding of information. This could be the case when, for example, a hand-written paragraph is scanned, and some of the characters are not well defined, as depicted in Fig. 2.
- *Accessibility:* Also called *availability*, is related to the ability of the user to access the information.
- *Trust:* Refers to how much the information source can be trusted, and therefore to what extent data are reliable. For example, people may rely on Twitter posts to find out the quality of a movie, or check the IMDB site at http://www.imdb.com, which might provide more reliable data.
- *Usefulness* (cf. Firmani et al. (2015)): This is related to the benefits a user can obtain when using the data to produce information. For example, Fig. 3 shows scans taken from Bosch's *The Garden of Earthly Delights*. To observe technical details present in the picture of this painting, a user would choose the image with the highest contrast. Again, this is a contextual quality dimension: a lower-quality picture may suffice for some users or for some kinds of requirements, while clearly not enough when the details are needed.

To quantify these dimensions and to be able to assess DQ according to them, the concept of *metrics* must be introduced. Mathematically, a DQ *metric* for a dimension D is a function that maps an entity to a value, such that this value, typically between 0 and 1, indicates the quality of a piece of data regarding the dimension D. For a given dimension, more than one metric could be defined and combined to obtain a concrete quality value. Note that metrics are highly context-dependent. For example, the readability of a hand-written text may be influenced not only by the text content, but also by the way the user writes. The same occurs with metrics for other DQ dimensions.

4 Big Data Quality

Since the present paper discusses DQ issues in a Big Data context, this concept is briefly addressed in this section. In a Big Data context, datasets are too large to store, analyze, handle or process, using traditional database tools.

Fig. 3 Usefulness issue example



As explained above, Big Data are characterized by "4 V's", namely *Volume* (size of the datasets), *Velocity* (speed of incoming data, e.g., the number of tweets per second (TPS)), *Variety* (refers to the type and nature of the data), and *Veracity* (the reliability of the data, which, in this context, is greatly volatile, even within the same data stream). In the literature, many other "V's" can be found, but only these four will be considered in the present work. Data can be classified, according their structure, as:

– *Stuctured*, where each piece of information has an associated fixed and formal structure, like in traditional relational databases;
– *Semi Structured*, where the structure of the data has some degree of flexibility (e.g., an XML file with no associated schema, or a JSON response from an API, whose structure is not completely defined);
– *Unstructured*, where no specific structure is defined.

Further, the United Nations Economic Commission for Europe (UNECE) classifies Big Data according to the data sources in: *Human sourced*; *Process mediated*; and *Machine generated* (The United Nations Economic Commission for Europe (UNECE)-Task Team on Big Data. Classification of types of big data 2007), defined as follows.

– *Human-sourced data:* Information that people provide via text, photos or videos. Usually, this information lacks of a fixed structure, like the texts written in natural language. Therefore, the information streamed here is *loosely structured* and often ungoverned. Data coming from social networks, like Twitter posts, YouTube videos, etc., are typical examples.
– *Process-mediated data:* Information that concerns some business events of interest, like the purchase of a camera in an e-commerce site or the sign-up of clients in a system. This information is *highly structured*, such as relational databases, coming from traditional Business systems.
– *Machine-generated data:* Refers to the data resulting from the tracking of sensors of the physical world (e.g., temperature sensors, human health sensors, GPS coordinates, etc.). This type of source is associated with very large amounts of data, given the constant tracking of the sensors. In general, these are data coming from the so-called Internet of Things.

Given these characteristics of Big Data, the DQ along the dimensions explained in Section 3 must be quantified using metrics specific to such a context, therefore the typical quality metrics used for structured, process-mediated data must be adapted to this new situation. This is studied in the next sections.

4.1 Data Quality Dimensions and Metrics in a Big Data Context

This section studies how the DQ dimensions can be used in a Big Data scenario. Metrics for the dimensions defined here are presented in the next section. It is worth noting that the study focuses on *human-sourced generated data*, since the aim is to address the quality of Twitter streams.

– *Readability* (r) Given a dictionary D, and a collection of words considered valid in a document x, the *Readability* of x, denoted $r(x)$ is defined as the quotient between the number of valid words in x and the number of words in x, if any, otherwise it is zero. That is, given a set W of the words (valid and non-valid) that are present in the document x, the readability of x is

$$r(x) = \begin{cases} \frac{|\{w \in W \ \wedge \ w \in D\}|}{|\{w \in W\}|} & if \ \ W \neq \emptyset \\ 0 & if \ \ W = \emptyset \end{cases}$$

In the remainder, the problem to be addressed will refer to tweets in a Twitter stream, thus x will represent a tweet.

– *Completeness* (c) Consider an object x in a domain, and an array $props_p$ that contains the names of the properties required to describe x for a given problem p; assume that x is represented as a collection of $(property, value)$ pairs of the form $\{(p_1, v_1), \ldots, (p_n, v_n)\}$, such that v_i is a value for p_i. If a property $p_i \in x$ has associated a non-null value v_i, it is called well-defined. Also, assume there is a function $validPropsOf(x, p)$ that, given an object x, and a set of properties $props_p$, returns the set of well-defined properties of x in $props_p$. The *Completeness* of x, denoted $c(x)$ tells to what extent are all the properties in $props_p$ well-defined in x. It is computed as:

$$c(x) = |validPropsOf(x, p)| \ / \ |props_p|$$

That means, c is a value between 0 and 1.

Example 1 (Completeness) Consider the tweet $x =$ {*text: "I like Bitcoin", user: null*}, and an array of required properties, $props_p = [text]$.

Given that the required property is present in the tweet, completeness is fully satisfied, thus $c(x) = 1$.

Consider now that the array of properties is $props_p = [text, user]$. In this case, since the *user* property has a null value, only half of the requirements are fulfilled by the tweet. Thus, $|props_p| = 2$, and $|validPropsOf(x, p)| = 1$, then $c(x) = 0.5$.

- *Usefulness (u)* Since this paper deals with human-sourced datasets, it will be assumed that this property is directly related to the possibility of (among others):
    - (a) Detecting a sentiment, whether positive or negative, in an object x, e.g., a tweet. Therefore, if x reflects a positive or negative feeling about a certain topic or person, x will be considered useful. If the sentiment is neutral, or no sentiment could be computed by a Natural Language Processing (NLP) tool, x will be considered not useful.
    - (b) Detecting the domain or topic of x, for example, politics, marketing, sports, and so on.

  There are of course many ways of assessing usefulness, but this is outside the scope of this paper. In the remainder, Usefulness is defined as:

$$u(x) = \begin{cases} 1 \ if \ \ (sentiment(x) = P \ \vee \ sentiment(x) = N) \\ 0 \quad otherwise \end{cases}$$

- *Trustworthiness (t)* In a social network (or, in general, for human-sourced datasets) anyone in general can publish any kind of information anywhere, whether truthful or not. With respect to DQ, this dimension is in general considered to be composed of three dimensions, namely believability, verifiability, and reputation (Firmani et al. 2015). Believability is a reference to the extent to which information can be considered credible. Verifiability refers to the degree by which a data consumer can assess the correctness of a data set. Reputation is a judgement made by a user to determine the reliability of a source. In a Twitter context, these three properties can be reflected by the reliability of the broadcaster of the tweet. In other words, to what extent the Twitter user can be reliable. In some sense, this can be measured by the number of followers of an account, the time elapsed since the account was created, and if the Twitter account corresponds to a user that has been verified or not. In this paper, for simplicity, Trustworthiness is defined as:

$$t(x) = \begin{cases} 1 & if \quad the\ user\ is\ a\ verified\ one \\ 0.5 & if \quad the\ user\ has\ more\ than\ X\ number\ of\ followers \\ & \qquad or\ more\ than\ Y\ days\ elapsed\ since\ the\ account \\ & \qquad was\ created \\ 0 & \qquad otherwise \end{cases}$$

4.2 Computing (Big) Data Quality

As discussed in the previous section, the definition of DQ is not the same for all contexts and problems, but normally it depends on them. This section provides a wide and general definition of DQ metrics in the context of Big Data, particularly instantiated for the Twitter case.

Definition 1 (Problem (p)) A problem p is a string or sequence of characters that defines the question to be solved in a human-readable way.

Example 2 (Problem) A problem can be defined as: *Given a Twitter stream, what are the best quality tweets for the hashtag #2020Elections?*.

Definition 2 (Domain Model (X)) A *Domain Model* is defined as the set of objects whose quality will be measured.

Example 3 (Domain Model) For the case that studied in this paper, the domain model is defined as *a collection of Twitter feeds*.

Definition 3 (Data Quality Metric (m_{Xp})) A *Data Quality Metric* is a function $m_{Xp} : X \rightarrow [0 \dots 1]$, such that, given $x \in X$, and a problem p, then $m_{Xp}(x) = 0$ if x contains data of very poor quality for the given problem p, and $m_{Xp}(x) = 1$ if x contains data of very good quality to fit the problem.

Definition 4 (Weight of a Data Quality Metric ($m_{Xp}.weight$)) The relevance of a metric m_{Xp} to address a problem p, is measured by a *weight* associated with such metric. This weight is a value between 0 and 1.

Definition 5 (Data Quality (Q_{Xp})) Consider a problem p, a domain X, and a set of metrics $M_{Xp} = \{m_1, m_2, ..., m_n\}$. Each m_i is a DQ metric function. Note that n is an integer number greater than zero and the set M_{Xp} is finite. *Data Quality* is a function $Q_{Xp} : X \rightarrow [0 \dots 1]$ such that $Q_{Xp}(x) = g_{(m_1,m_2,\dots,m_n)}(x)$, where g is a function $g : (X \rightarrow [0, 1])^n \rightarrow (X \rightarrow [0 \dots 1])$.

The quality of a tweet x will be computed as $Q(x) = g_{(r,c,u,t)}(x) = \sum_{m=\{r,c,u\}} m(x) * m.weight$, where r, c, u and t are, respectively, the *Readability*, *Completeness*, *Usefulness*, and *Trustworthiness*, defined in Section 4.1.

Example 4 (Data Quality of a Tweet) Consider the following tweet, and, generically call it x:

```
-    text:``I love Big Data Quality
m#a!sc["
-id: 1
-coordinates: [48.864716, 2.349014]
```

Also consider the set $props_p = \{text, id\}$, and the weights $r.weight = 0.25$, $c.weight = 0.25$, $u.weight =$

0.25, and $t.weight = 0.25$, for each of the DQ dimensions in Section 4.1. The quality of x is computed as follows.

- *Readability(r)*

  $$r(x) = \frac{|\{I, love, Big, Data, Quality\}|}{|\{I, love, Big, Data, Quality, m\#a!sc[\}|}$$

  $$r(x) = \frac{5}{6} = 0.833$$
- *Completeness(c)* Given that $props_p = \{text, id\}$, then:

  $c(x) = |validPropsOf(x, p)| \;/\; |props_p|$

  Since $props_p = \{text, id\}$, and $validPropsOf(x, p) = \{text, id\}$,

  $c(x) = 2 \;/\; 2 = 1$
- *Usefulness(u)* The text provided expresses positive sentiment, thus:

  $sentiment(x) = P,\ and\ u(x) = 1.$
- *Trustworthiness (t)* Considering that the tweet is posted by a verified user, $t = 1$ is assumed.

Finally, the quality value for x is $Q(x) = 0.83 * 0.25 + 1 * 0.25 + 1 * 0.25 + 1 * 0.25 = 0.9575$.

4.3 Accounting for Context

Example 4 assumes that all metrics have the same weight. However, analysts may want to weight differently the quality dimensions, according to their interests. For instance, in Example 4, the user may be more interested in trustworthiness than in the other dimensions, thus, she would decide to give a higher weight to the former (e.g., 0.55), keep the weight for readability, and reduce the weight of the other two (0.1 for each one). The quality value for x would be, in this case:

$Q(x) = 0.83*0.25+1*0.1+1*0.1+1*0.55 = 0.9575.$ In case the user is more interested in readability, she would give a higher weight to this dimension, reducing the weight of the other ones, for example, 0.4, 0.2, 0.1, and 0.2. The quality value for x would be, in this case:

$Q(x) = 0.83 * 0.4 + 1 * 0.2 + 1 * 0.15 + 1 * 0.2 = 0.882.$

The example depicted in Figs. 4 through 6 in Section 4.4, showed the average data quality when looking for the keywords “coronavirus vaccine”, considering a weight of 0.25 for each dimension quality metric, returning values of 0.89 and 0.82, for the overall data quality in each case. The system allows changing dynamically those weights. For the values in the example above (i.e., 0.25, ,1, 1, 0.55), the general data quality resulted in this case, of 0.98 and 0.866, that means, higher.

4.4 A Real-World Example

The following example wraps up the above, with the help of the tools that are described in the next section. Nowadays, people are posting an enormous number of comments about the coronavirus pandemic. Therefore, it is crucial for decision makers to assess the reliability and quality of these posts. Figure 4 depicts the data quality results, obtained in real time, for the tweets containing the keywords “coronavirus vaccine”. The figure on the left-hand side shows that the data quality ($Q(x)$ from Definition 5) of most tweets is in the 0.8-1 range. The quality dimensions used are the ones defined in Section 4.1. The overall data quality is 0.82, computed using, for the computation of the trustworthiness quality dimension, a threshold of 120 days since registration and 100 followers; the figure on

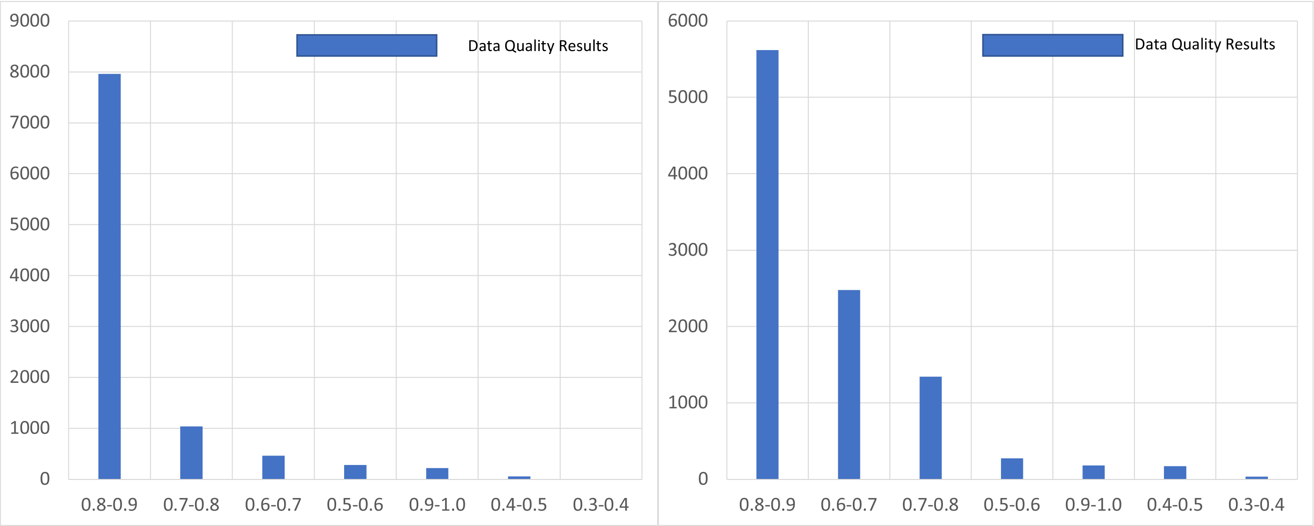


Fig. 4 Overall data quality of tweets including “coronavirus vaccine”

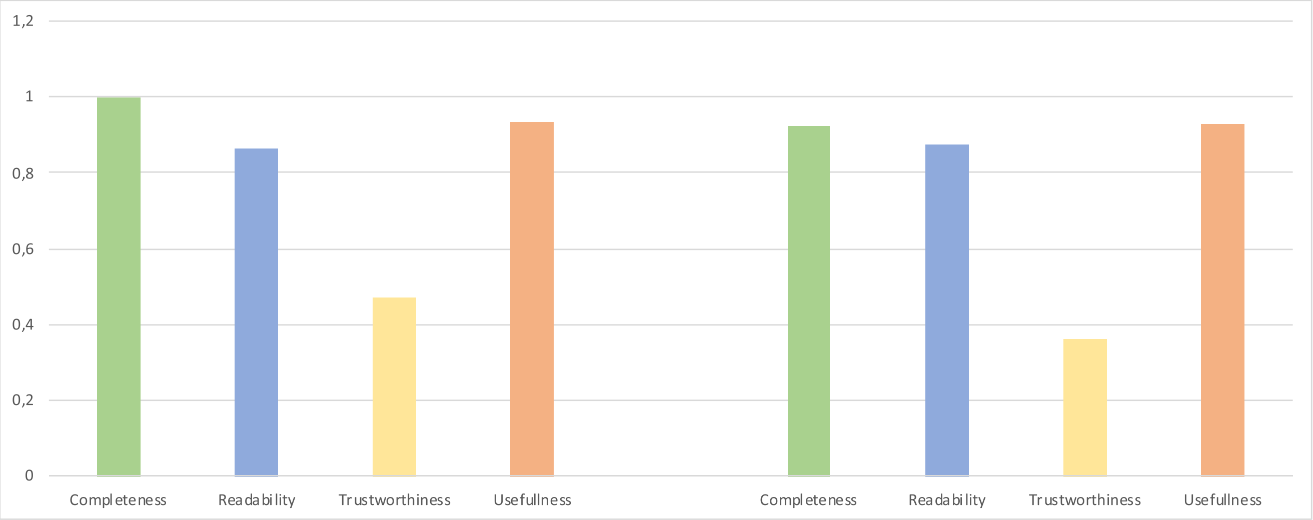


Fig. 5 Data quality for each dimension, for tweets including the kewyword "coronavirus vaccine"

the right-hand side shows the results using the values 60 and 10, respectively, for the former parameters. The overall data quality in this case is 0.77, lower than the previous one, because of the higher trustworthiness requirement. Note also that the number of tweets in the 0.6-0.7 range is higher than in the previous case. Figure 5 shows each quality dimension metric separately. It can be seen that trustworthiness and completeness have lower quality for the case where the thresholds are higher (i.e., the right-hand side of Fig. 4). Figure 6 shows the general results as displayed by the tool that will be explained in Section 5. The tool allows to directly compare the results of both streams simultaneously. As additional information, the most relevant hashtags in the stream are also displayed for both cases.

5 Implementation

This section presents and describes the implementation of the concepts previously explained. The architecture is described first, detailing the technological components and how they interact with each other for capturing, filtering, computing data quality, and displaying the results. Finally, the user interface (UI) is described. The goal of the implementation is to develop a system that can let users capture a collection of tweets from a stream of Twitter feeds, based on a particular keyword search. The system must also compute the quality of the data, and visualize the results to gain insight on such quality.[2] Figure 7 shows the general scheme of the system, and its three parts: ingestion, transformation and visualization.

The core of the ingestion part is an Apache Kafka[3] instance. Kafka is a distributed streaming platform for capturing, processing and storing data streams in real time. A Zookeeper[4] service manages the message topics[5] and, in case Kafka is installed in a cluster, coordinates the cluster nodes. Besides the Kafka core, there are components for producing, consuming, storing and displaying data, as well as a mail service to communicate with the users (e.g., for security management). Figure 8 illustrates these components and their orchestration, together with a description of the actions that occur when a keyword-based search is initiated by a user. A more detailed explanation is given next.

The process is as follows. The endpoint for the users is located at the URL http://dataquality.it.itba.edu.ar. From the web UI, the user starts a keyword-based search that sends a POST request to the REST API. The Proxy serves the request and redirects it to the Kafka producer service, which starts the session with Twitter, and publishes to a particular topic (a 32-bytes identifier) the messages that it finds. Each time Twitter finds a result, sends it in real time to the Kafka producer service, which computes the data quality values. Then, the microservice which serves the *consumer* service at the Proxy, receives the events queued at the Kafka topic, and stores the data in a new ElasticSearch[6] index. In addition, the UI component makes a request to the REST API, which is redirected to ElasticSearch through a query.

[2] The implementation is available at http://dataquality.it.ita.edu.arb/, and can be used with credentials usertest/usertest.

[3] https://kafka.apache.org/

[4] https://zookeeper.apache.org/

[5] Kafka records are organized into topics, such that a Kafka topic is a feed name to which records are stored and published. Producer applications write data to topics and consumer applications read from topics.

[6] https://www.elastic.co/elasticsearch/

That means, the searches and their results are stored in ElasticSearch, so the user may retrieve them whenever she wants. The REST API can also add data based on the user preferences. Figure 8 depicts the data flow in the system, which starts with a user's request on the web interface (Step 1). The request is sent to the API (Step 3), which forwards it to the Kafka producer, which in turn negociates with the Twitter API (Steps 5, 6, 7). The results are sent to the Kafka broker where the *consumer* service takes the data (Steps 8,9). Steps 10 and 11 show how data are persisted in, and retrieved from ElasticSearch.

The REST API has been developed using NodeJS v10.16.3 (Nodejs & express. OpenJS Foundation 2020), and the Express.js framework[7]. In addition, other external modules are used for communication with PostgreSQL and ElasticSearch, and mail services are used to communicate with the user. The API software design follows the Model-View-Controller pattern, where each route is mapped to a particular controller, which is associated with one or more models. The class model is depicted in Figure 9. The user entry points are the "routes". Each route can execute one or more controllers. Each controller talks with its corresponding model, which is an object-relational mapping (ORM) to the PostgreSQL database. This is done through the Sequelize framework[8], the ORM support in NodeJS. This framework translates a JSON-based syntax to SQL. This architecture would make it easy to migrate to any relational database, if needed, without rewriting code. For user authentication, JSON Web Tokens[9] are used. The verification process is run by the REST API using these tokens. This is not only used for login, but also for all routes. The routes are split in various segments. The main ones make requests to the *ElasticController*, the database controllers and the controller that calls the Proxy consumer. According to the Model-View-Controller pattern, not all routes contain the business logic, but call one or more controllers. All controllers have two input parameters: (a) the Topic ID, which is the UUIDv6 of the search topic, and is used to execute the queries to the correct ElasticSearch index; (b) the CallbackOK function, which is the function to be executed if the answer is valid and complete. An optional parameter indicates the function to be ran in case of error.

Since the UI plays a key role in this paper, it is described in more detail next. The UI is built on top of the dc.js library, which uses the crossfilter.js framework, and the data visualization library d3.js. Interactive maps were developed with the Leaflet library[10], and charts with the Chart.js[11]. The UI is composed of two main parts: one, to define the searching parameters, keywords, filters, the weight of each DQ metric, and other features; the other one, to display graphically the quality results as data are ingested. That means, the graphics become updated with new data at regular intervals. Figure 10 shows the part of the UI used to enter the different parameters (the part of the UI that displays the results is shown in the next section). In the box at the top, a list of search keywords can be defined, to filter out the tweets that do not contain at least one of them. There is a tag indicating if only tweets written in English will be considered. There is also a list of fields that the user checks, to define the schema of properties to compute the Completeness dimension metric. The "keywords" box allows entering a list of keywords that must be present in a tweet included in the computation of the DQ. Then, the parameters to compute trustworthiness are defined, as well as the weights for each of the four DQ dimensions. In the vertical menu on the left hand side, other sub-menus are available, allowing: (a) querying historical searches, and also pairwise comparing them, as shown in Fig. 6; (b) defining topics such that if a tweet contains words from a list, the system will classify the tweet as belonging to a certain topic. Even this is a very simple classification, it suffices for the goals of the experiments presented in the next section; (c) defining a user profile; (d) adding new users.

Figure 11 shows the screen allowing analyzing past searches. The "Analyze Dashboard" button takes the user to the standard dashboard, which assesses data quality using the fixed weights. The "Analyze Dashboard Context" button takes the user to a menu that allows to dynamically change the weights of the quality metrics, according to the analysis context. The "Compare" menu allows opening together two historical dashboards, and compare the results as shown in Fig. 6.

The UI computes and displays the following:

- (a) A histogram with the general DQ of a stream, in a given time interval.
- (b) A detail of the value of each DQ dimension.
- (c) The same as (b), distinguishing between original and retweeted tweets.
- (d) The same as (b), distinguishing between tweets posted by verified and unverified users.
- (e) The DQ of the tweets corresponding to different (user defined) topics.
- (f) A map where the quality of the geolocalized tweets is displayed, allowing analyzing the DQ of tweets with respect to the geographical locations.

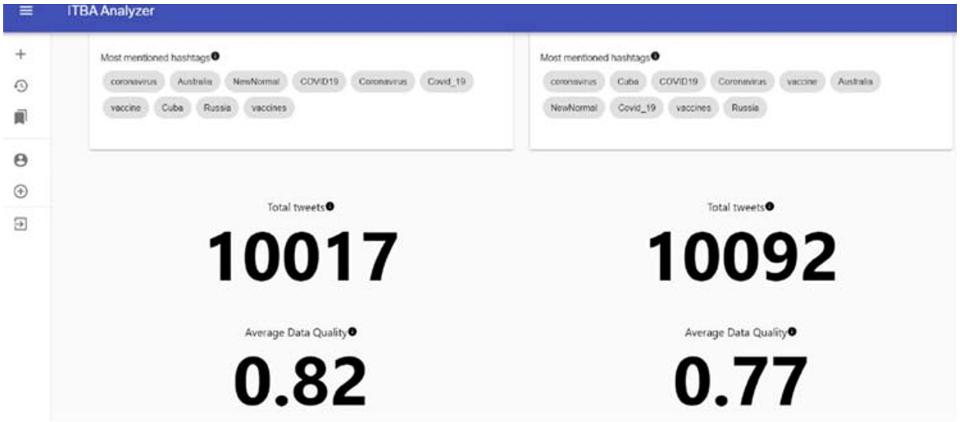
[7] https://github.com/expressjs

[8] https://sequelize.org/

[9] https://jwt.io/

[10] https://leafletjs.com/

[11] https://www.chartjs.org/



ITBA Analyzer

Most mentioned hashtags

coronavirus Australia NewNormal COVID19 Coronavirus Covid_19 vaccine Cuba Russia vaccines

Total tweets

10017

Average Data Quality

0.82

Most mentioned hashtags

coronavirus Cuba COVID19 Coronavirus vaccine Australia NewNormal Covid_19 vaccines Russia

Total tweets

10092

Average Data Quality

0.77

Fig. 6 Comparing data quality of two Twitter streams including the kewyword "coronavirus vaccine", using the analytical tool

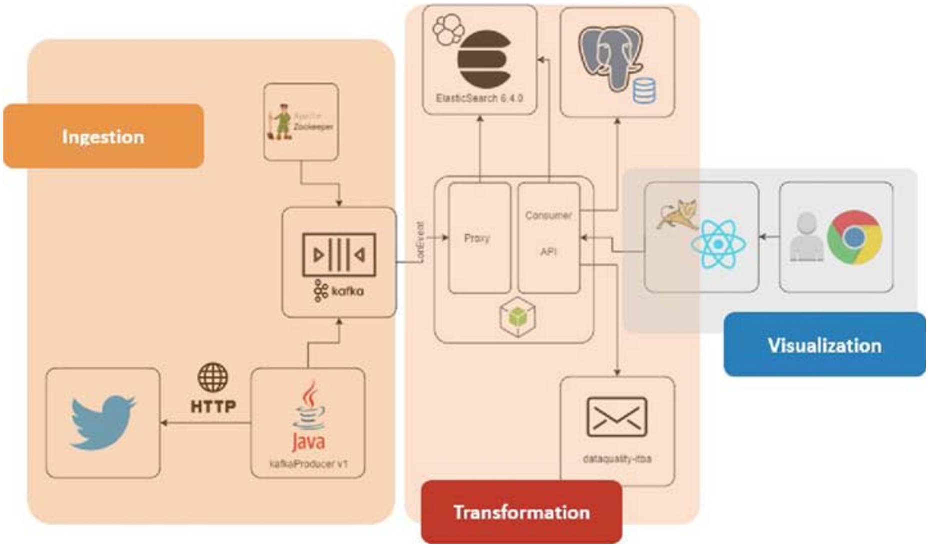


Fig. 7 General scheme

The API Rest queries Elastic Search (11). When the first tweet arrives, an index is created

The Proxy detects the event and sends it to Elastic Search

4. Stores the search in the database

The topic is created

2. Search template loaded

1. User goes to dataquality.it.itba.edu.ar

5. Data sent to the kafkaProducer

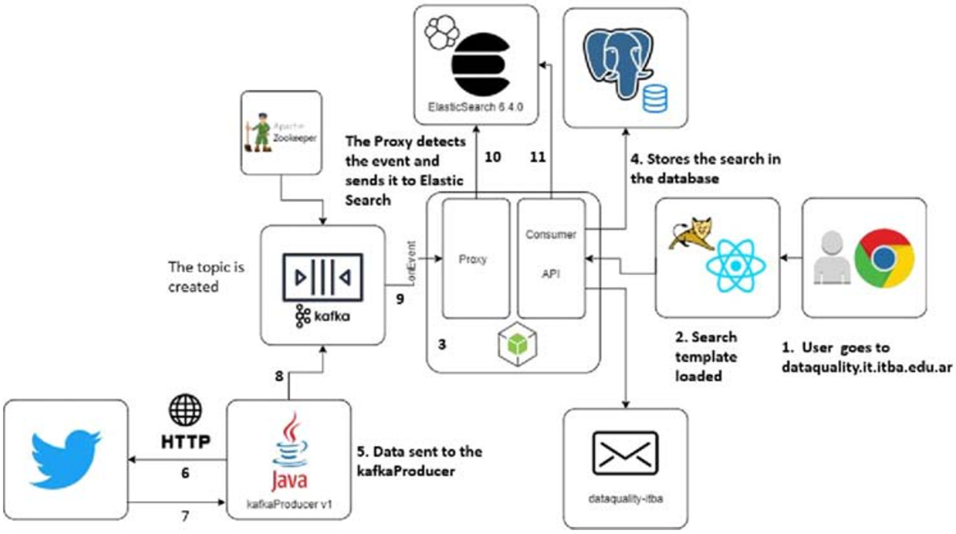


Fig. 8 Architecture

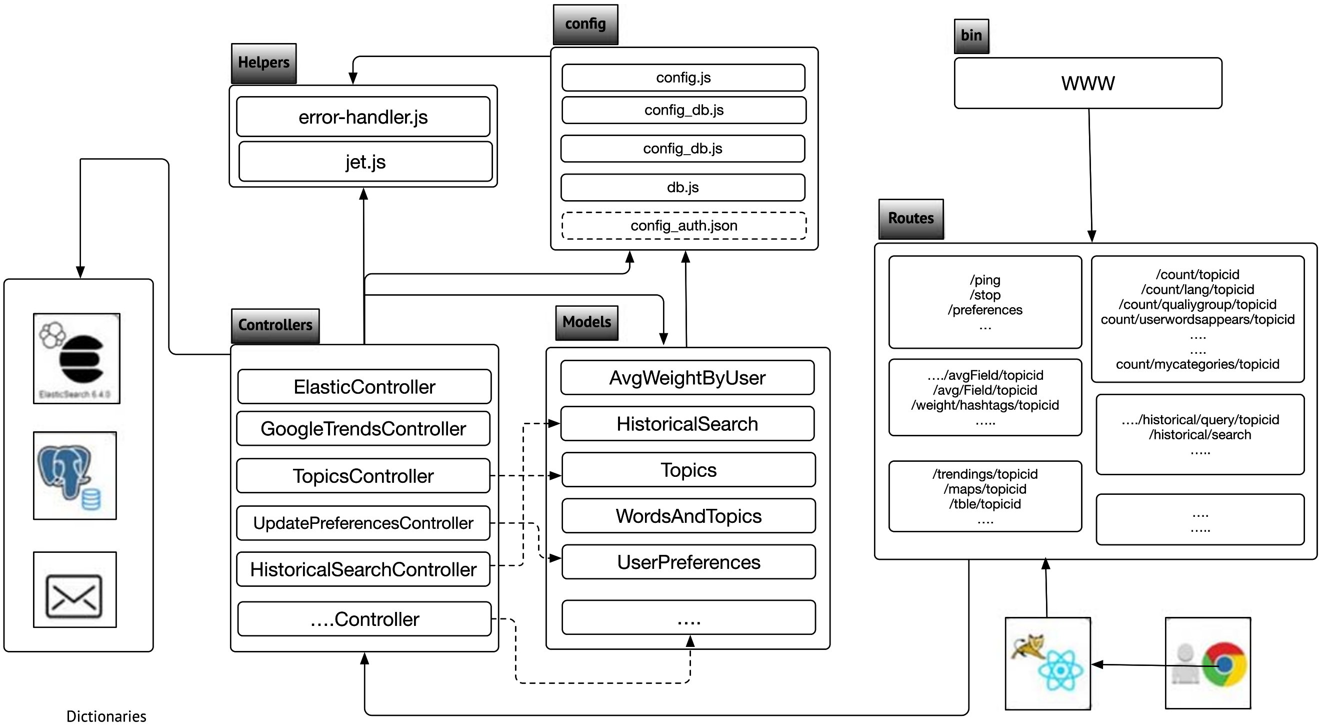


Fig. 9 Class Model

6 A study of the Quality of Twitter Data

This section describes the experiments performed over the implementation presented in Section 5, reports the results, and discusses them. The goal of the experiments is to illustrate how the quality of a Twitter stream can be assessed using the dimensions and metrics presented in the previous sections, and using the software described in Section 5. Thus, the report that follows will show the features of the tool presented in this paper.

The quality dimensions considered in all cases are: readability, completeness, usefulness, and trustworthiness, with the metrics described in Section 4.1, and with an initial weight or 0.25 for each dimension metric. Nevertheless, the tool allows the user to modify the weights according to her analysis needs. The dictionary used to check readability is given in English-words project (2018) and contains 479,000 english words, including acronyms, abbreviations and even Internet slang. To compute sentiment (for usefulness), the Stanford CoreNLP software was used (Manning et al. 2014). In all cases, the following features are computed and displayed (other analytical features are shown later):

- Overall DQ of the stream.
- The values of each quality dimension individually.
- DQ dimension values (general and by dimension) corresponding to verified and unverified users.
- DQ dimension values (general and by dimension) grouping tweets by topic.
- DQ of the tweets that contain the geographical coordinates from where they were posted.
- Geolocalization in a map, of tweets containing coordinates.

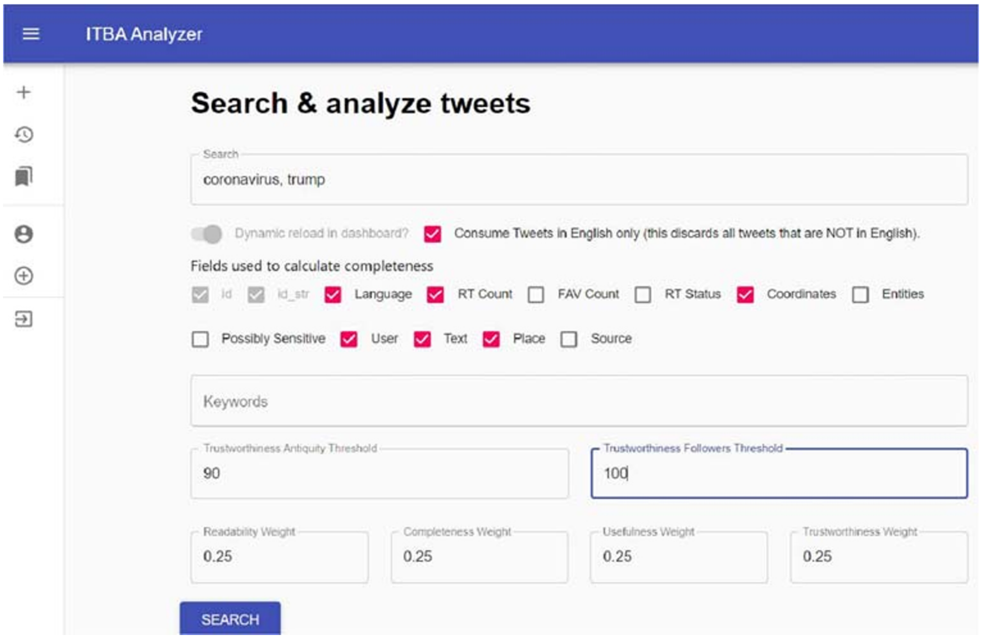
The study performed over Twitter data is aimed at providing evidence on the general quality of these data. The general belief is that Twitter data is unreliable, disperse, and of limited use for decision making. The study is aimed at confirming or denying this assumption. The study is presented in a classic way. First, the hypotheses are stated (Section 6.1). Then a collection of tests associated with those hypotheses are defined and the tests are ran in a certain environment (Section 6.2). Finally, results are reported and discussed (Section 6.3).

6.1 Problems and Hypotheses

The goals and hypotheses defined for the experiments are described next.

- *Problem 1*. Compare the overall DQ of the whole stream of tweets, against the DQ of a stream filtered by a set of keywords. The hypothesis is that the latter are more likely to have better quality than the former, since they are focused on a "controlled" subset.

Fig. 10 Main User Interface



Search & analyze tweets

Search: coronavirus, trump

Dynamic reload in dashboard? ☑ Consume Tweets in English only (this discards all tweets that are NOT in English).

Fields used to calculate completeness

☑ id ☑ id_str ☑ Language ☑ RT Count □ FAV Count □ RT Status ☑ Coordinates □ Entities

□ Possibly Sensitive ☑ User ☑ Text ☑ Place □ Source

Keywords

Trustworthiness Antiquity Threshold: 90

Trustworthiness Followers Threshold: 100

Readability Weight: 0.25

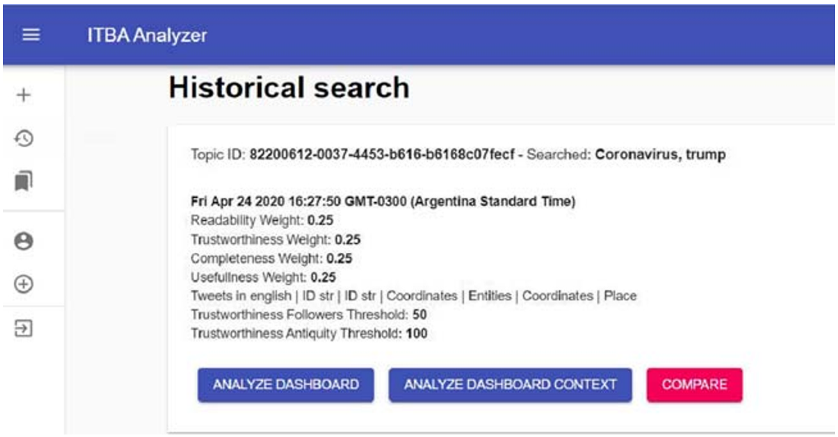
Completeness Weight: 0.25

Usefulness Weight: 0.25

Trustworthiness Weight: 0.25

- *Problem 2.* Same as above, for each of the four quality dimensions considered, in order to study if there are differences across the quality dimensions.
- *Problem 3.* Compare the quality of tweets posted by verified and not-verified users. Here the hypothesis is that verified users post higher quality tweets, not only considering the higher value of the Trustworthiness dimension, but also leading to higher values in other dimensions.
- *Problem 4.* Compare the DQ of tweets referring to different topics. The intention here is to determine if there are topics whose corresponding tweets have a DQ that is, in general, better than the one of some other topics. For example, *a priori* one may guess that tweets referring to politics may have higher quality than tweets referring to sports.
- *Problem 5.* Although the percentage of geolocalized tweets is very low compared to the total number of tweets, the study also looks for countries/regions/cities that could be characterized by a better DQ, and also study if users from different regions are more likely than other ones, to open their geographic location.
- *Problem 6.* Determine the influence of the ”age” of an account, and the number of followers, in the value of the Trustworthiness dimension. The question is: are tweets from older accounts and posted by users with a higher number of followers, of higher quality than the rest?
- *Problem 7.* Study the impact of completeness in DQ. For this, the same comparisons above are also performed requesting the presence of different sets of properties in order to compute completeness (that is, changing the requested schema). That is, if

Fig. 11 Search history



Historical search

Topic ID: **82200612-0037-4453-b616-b6168c07fecf** - Searched: **Coronavirus, trump**

Fri Apr 24 2020 16:27:50 GMT-0300 (Argentina Standard Time)
Readability Weight: **0.25**
Trustworthiness Weight: **0.25**
Completeness Weight: **0.25**
Usefulness Weight: **0.25**
Tweets in english | ID str | ID str | Coordinates | Entities | Coordinates | Place
Trustworthiness Followers Threshold: **50**
Trustworthiness Antiquity Threshold: **100**

Table 1 Number of tweets for each test

Test	# tweets
$test_1$	72,228
$test_2$	82,482
$test_3$	74,178
$test_4$	11,075
$test_5$	11,876
$test_6$	33,269
$test_7$	7,338
$test_8$	9,846

completeness is lower, does this impact on the other dimensions?

6.2 Defining and Running the tests

To address the problems above, the following tests were performed. Except when noted, the set of required properties for computing completeness is: $props_{p_1} = \{\texttt{id}, \texttt{id\_str}, \texttt{lang}, \texttt{retweet\_count}, \texttt{usr}, \texttt{text}, \texttt{source}\}$. Also, except for the second test, for *Trustworthiness*, the parameters were set to 100 followers, and an account created at least 120 days ago. The description of the tests is given next.

Test 1 The first test, t_1, analyzes the whole stream of tweets, with no keyword filtering, i.e., considering all tweets returned by the Twitter API.

Test 2 Test t_2 is aimed at analyzing the impact of Trustworthiness on the DQ results. Thus, the parameters used were set to 50 followers, and an account created at least 10 days ago. As in t_1, no keyword filtering was applied.

Test 3 Test t_3 considers only tweets containing at least one keyword from a collection. These keywords are defined as the union of all the keywords defining the four topics that are considered in this study: Politics, General Interest, Arts, and Sports. Among these keywords, the following terms are defined: {`Trump`, `#Politics`, ..., `#art`, `#movies`, ..., `#sports`}.

Test 4 Test t_4 analyzes tweets related to *political issues*. The assumption here is that, if a tweet includes hashtags like `#Politics`, `#Trump`, etc., they are likely to be referring to politics. Of course, more sophisticated topic discovery techniques could be used, but this may affect performance. In addition, the precision obtained with this simplified method is enough for the goals of this experiment.

Test 5 Test t_5 analyzes tweets related to sports. The assumption is that, if a tweet includes hashtags like `#sports`,`#soccer`, `#baseball`, etc., they are likely to be referring to some sport subject.

Test 6 Test t_6 studies only tweets mentioning the hashtag `#Coronavirus`, highly used at the time of writing this paper. Although this can be considered a particular case of a stream of political tweets, the idea is to investigate if this particular subset presents some distinguished characteristics.

Test 7 In Test t_7, all the tweets in a stream like in Test t_1 are considered, but the schema contains all the properties in $props_{p_1}$, plus the geographic coordinates, to analyze if the fact that people include the geographic coordinates in a tweet has an impact on the data quality.

Test 8 Test t_8 analyzes all the tweets in a stream like in Test t_1, but considering a schema containing all the properties supported by the UI, and comparing the results against the same test, but requesting properties in schema $props_{p_1}$.

Tweets were captured and displayed at a rate of 1000 per minute (for non-filtered tweets), and at about 200 per minute, for filtered tweets, depending on how many tweets pass the filters. For example, during the sports lockout season due to the pandemic, the frequency of sports-related tweets was clearly lower than in normal times. Since it was observed that after some number of tweets the DQ results become stabilized, tests were stopped when this convergence was reached. In some cases, however, the number of tweets was allowed to grow, to show that the tool can handle higher volumes.

The results obtained for the problems above are commented next, and (partially, due to space restrictions) illustrated by graphics produced by the UI of the software. Table 1 shows, for each test, the number of tweets processed, and the number of retweeted tweets included in each case. Note that for Tests 4 through 8, these numbers do not reflect the actual number of tweets acquired, but the number of tweets processed (the ones belonging to a certain topic). The analysis below is divided according to the problems that each test (or group of tests) addresses. The problems' descriptions are repeated to facilitate reading.

6.3 Results and Discussion

Problems 1 and 2 : Compare the general quality of the whole stream of tweets, against the quality of a stream filtered by a set of keywords related to different topics, and also study the DQ of each of the four dimensions considered. Tests 1 through 3 address this problem. Figure 12 depicts the general data quality for the two cases, for Tests 1 and 3, using the capability of the tool to compare two searches. The overal data quality of the stream is 0.7

for Test 1, and 0.71 for Test 3. However, it can be seen that filtering the streams using some keywords increases the DQ of the data obtained, in the following sense: In Test 1, the DQ range with most tweets is 0.6-0.7, while in Test 3 the first range is 0.7-0.8; also, the second range is 0.7-0.8 for not-filtered data, and 0.8-0.9 for filtered data. The third position is 0.8-0.9 for not-filtered data, and 0.6-0.7 for filtered data. Thus, clearly, filtering tweets through keywords increases the DQ of the retrieved stream.

Problem 3: Compare the quality of tweets posted by verified and not-verified users. All tests address this problem. Figure 13 depicts the DQ dimensions for Tests 1 and 3, considering tweets in the stream coming from verified users, against tweets coming from unverified users. It can be seen that DQ dimensions deliver better results for tweets coming from verified users, although this difference becomes lower for filtered tweets. Because of its definition, Usefulness has value "1" for all tweets coming from verified users.

Problem 4: Compare the DQ of tweets referring to different topics. The intention here is to determine if there are topics whose corresponding tweets have a DQ that is, in general, better than the one of some other topics. Tests 4 through 6 address this problem. Figure 14 depicts the results produced by these tests. For reasons of brevity only the results for the overall DQ of the data streams are displayed. The general averages of the stream are similar in both cases (0.7). In the figure, tweets related to sports are compared against tweets related to politics. It can be seen that in the case of tweets related to sports (on the right), the third position is occupied by the tweets in the interval 0.5-0.6. In the case of politics, this position is occupied by the tweets in the interval 0.8-0.9, indicating a lower DQ. However, if the comparison is made against the whole stream, the DQ of sports-related tweets is similar.

Problem 5: Characterize DQ in terms of the geolocalization of the posting place. It is a well-known fact, acknowledged by the Twitter Company, that only less than 2% of the tweets include the coordinates from where it was posted. This was confirmed by the tests. These tweets where analyzed, aimed at concluding if there is a correlation between geographic regions and the DQ of the tweets. In Fig. 15 two characteristics can be noted: on the one hand, most of the geolocalized tweets come from the east and west cost in the US (mainly from the former), and from India; on the other hand, tweets from the east coast show, in general, better quality, and tweets from India have lower data quality than the former.

Problem 6: Determine the influence of the "age" of an account, and the number of followers, in the value of the Trustworthiness dimension. This problem involves tests 1 and 2. In the first case, the parameters were set to request 100 followers or a minimum of 120 days-old account, to qualify as a "1" for Trustworthiness (which are the parameters used in all tests except from Test 2). For Test 2, these parameters were lowered to 50 followers and 10 days-old accounts. Figure 16 depicts the results, using the "Compare" feature of the UI. The general DQ is 0.7 for Test 1 and 0.72 for Test 2. It can be seen that although the values of Trustworthiness are lower for the higher values (this is the reason for the higher average DQ of Test 2), this does not have a relevant impact over the other parameters. However, it is worth remarking that the distribution is different: the difference between the number of tweets in the third range is larger in Test 2. Nevertheless, in this case, no definitive conclusion can be drawn regarding the impact of considering tweets from more recent users.

6.4 Other Features

The UI allows other kinds of analysis, not included in the experiments, for the sake of space. These features include (this is an incomplete list, the user can check at http://dataquality.it.itba.edu.ar):

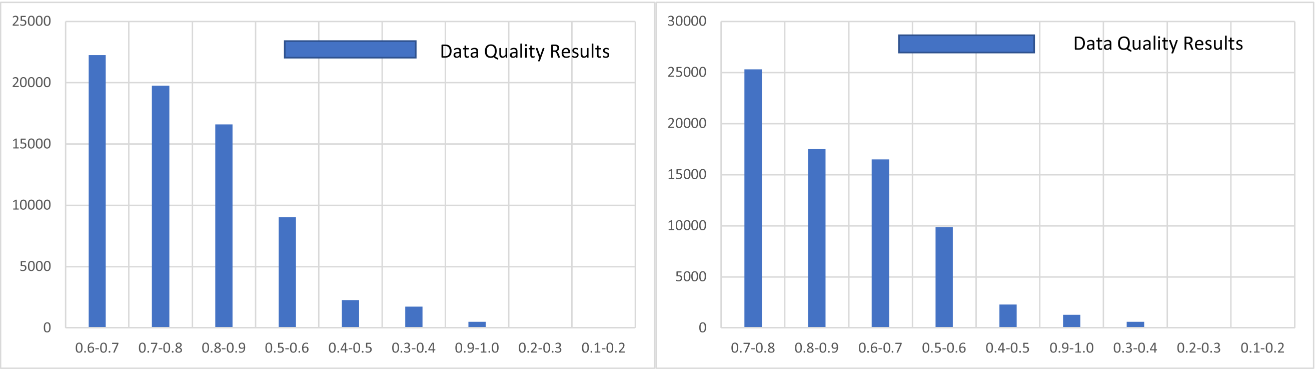


Fig. 12 Results for Problems 1 and 2 (General DQ). Left: Not-filtered stream; Right: Keyword-filtered stream

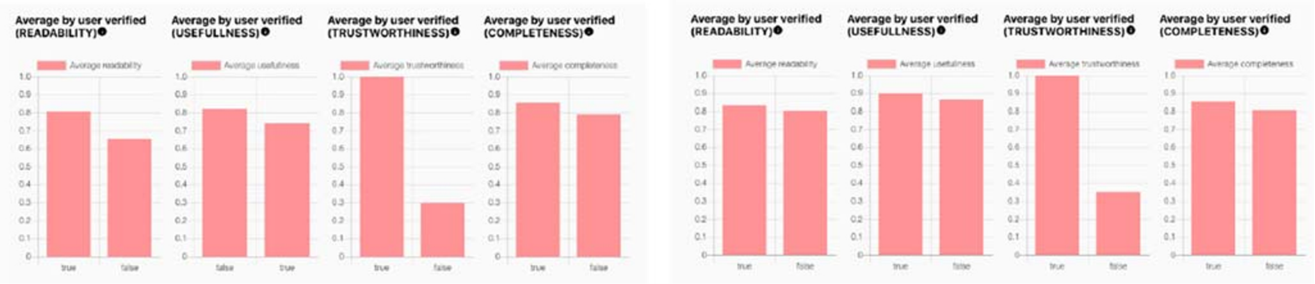


Fig. 13 Results for Problem 3 (Verified vs. unverified users). Left: Not-filtered stream; Right: Keyword-filtered stream.

- Graphics for DQ by keyword and by hashtag (see Fig. 17)
- DQ and text for the most re-tweeted tweets (see Fig. 18)
- Pairwise comparison of searches

Finally, results can be analyzed varying the weights of the DQ dimensions. The default used throughout the experiments is to evenly distribute weights. However, the tool allows to change these weights. For example, a user may give zero to Usefullness and Trustworthiness, 0.25 to completeness and 0.75 to readability. For Test 2 this gives an average of 0.82. Also, results are different when analyzing data in more detail. Figure 19 shows the average data quality discriminated by topics selected by the user. It is clear that both distributions are different when only two dimensions are considered, reflecting, in this case, readability in a stronger way.

7 Conclusion and Future Work

Given that social networks and Twitter in particular, are increasingly becoming a source of data for information systems of many different kinds, the need for assessing the quality of data obtained from these sources is crucial to guarantee that those systems will be provided of reliable and useful data. In spite of the relevance of this problem, the paper showed that this problem has not been addressed so far. Therefore, most works using these data assume that the quality of social media data is high enough for the goals pursued, without any guarantee that this is actually the case. In light of the above, the main question that this paper tackled was: “*Can Twitter data be used, reliably, to help in the decision-making process?*”. The work presented here aims at answering this question by means of performing a thorough study of the quality of Twitter data, using classic data quality concepts borrowed from the relational database world, adapting them to a Big Data scenario. Thus, the most typical quality dimensions (Readability, Completeness, Usefulness, Trustworthiness) were redefined in terms of the elements present in Twitter feed streams, as well as the corresponding metrics. Also, to automate this task, new software was developed to acquire and process streams of tweets in real time. The software includes a web user interface that allows the user to define input parameters used for the computation of the quality metrics. This interface displays the results in real-time, and also allows storing past searches using ElasticSearch indexing. Once this theoretical and technological infrastructure was set up, experiments were run to answer the main research question. These experiments aimed at computing the data quality of Twitter feeds for several different cases: a whole stream, a stream filtered with keywords corresponding to different topics, etc. In addition, taking advantage of the functionalities of the user interface, the different quality

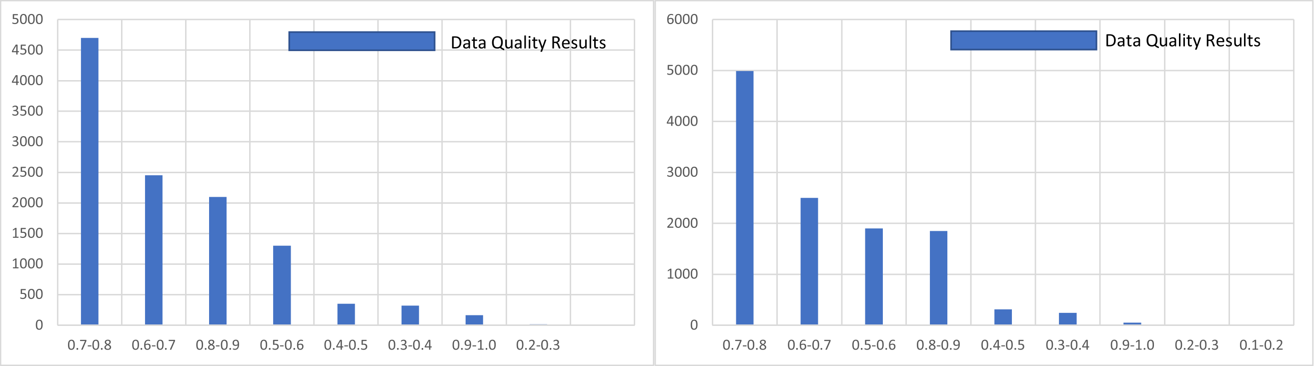
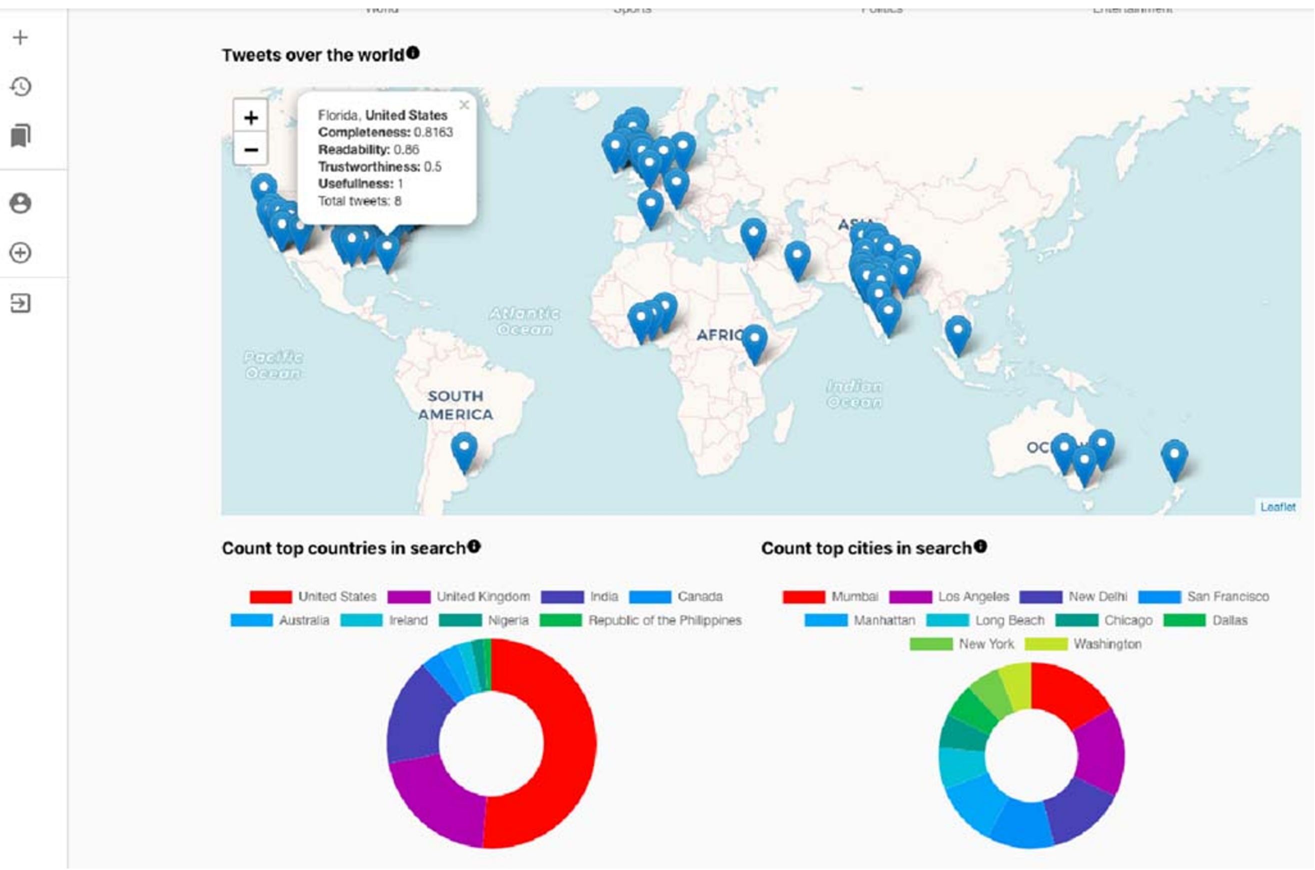


Fig. 14 Results for Problem 4 (DQ per topic - general). Left: politics-related tweets; Right: sports-related tweets



Florida, **United States**
Completeness: 0.8163
Readability: 0.86
Trustworthiness: 0.5
Usefullness: 1
Total tweets: 8

Fig. 15 Results for Problem 5 (Geolocalization-related DQ)

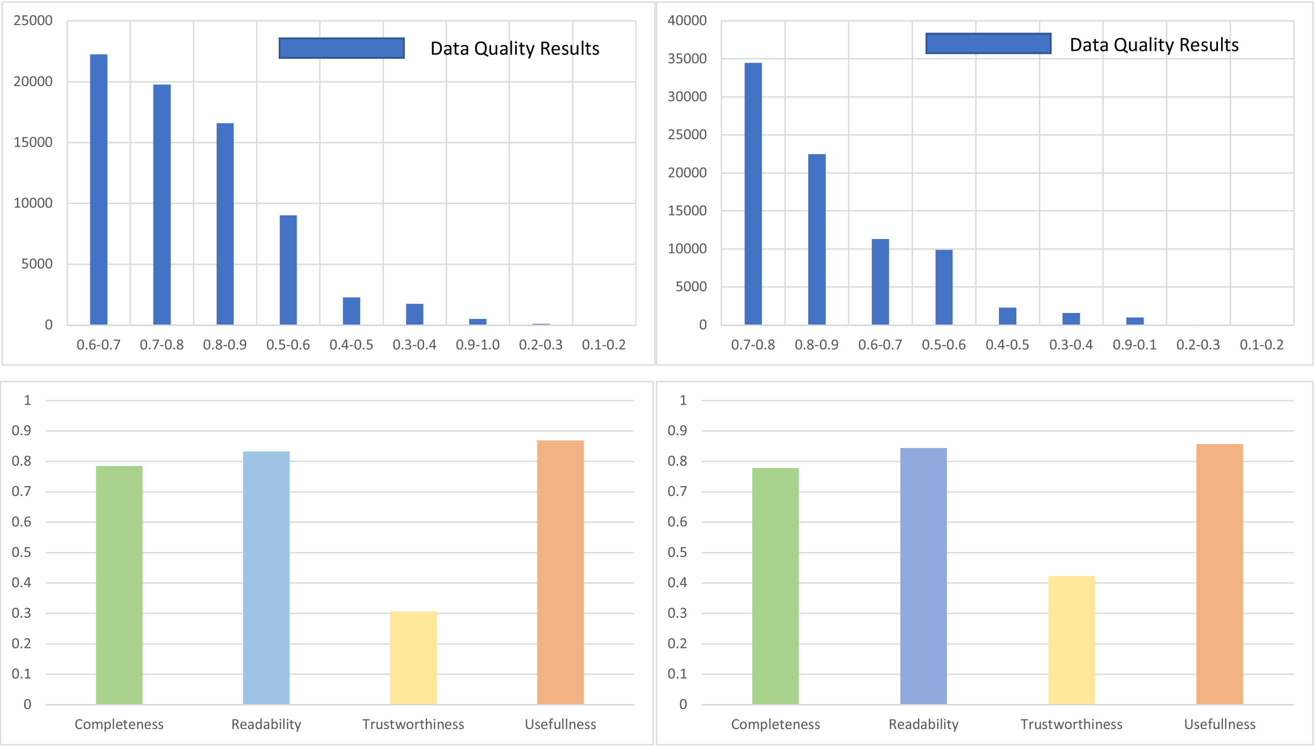


Fig. 16 Results for Problem 6 (Influence of Trustworthiness parameters). Left: 100 followers - 120 days-old accounts; Right: 50 followers - 10 days-old accounts

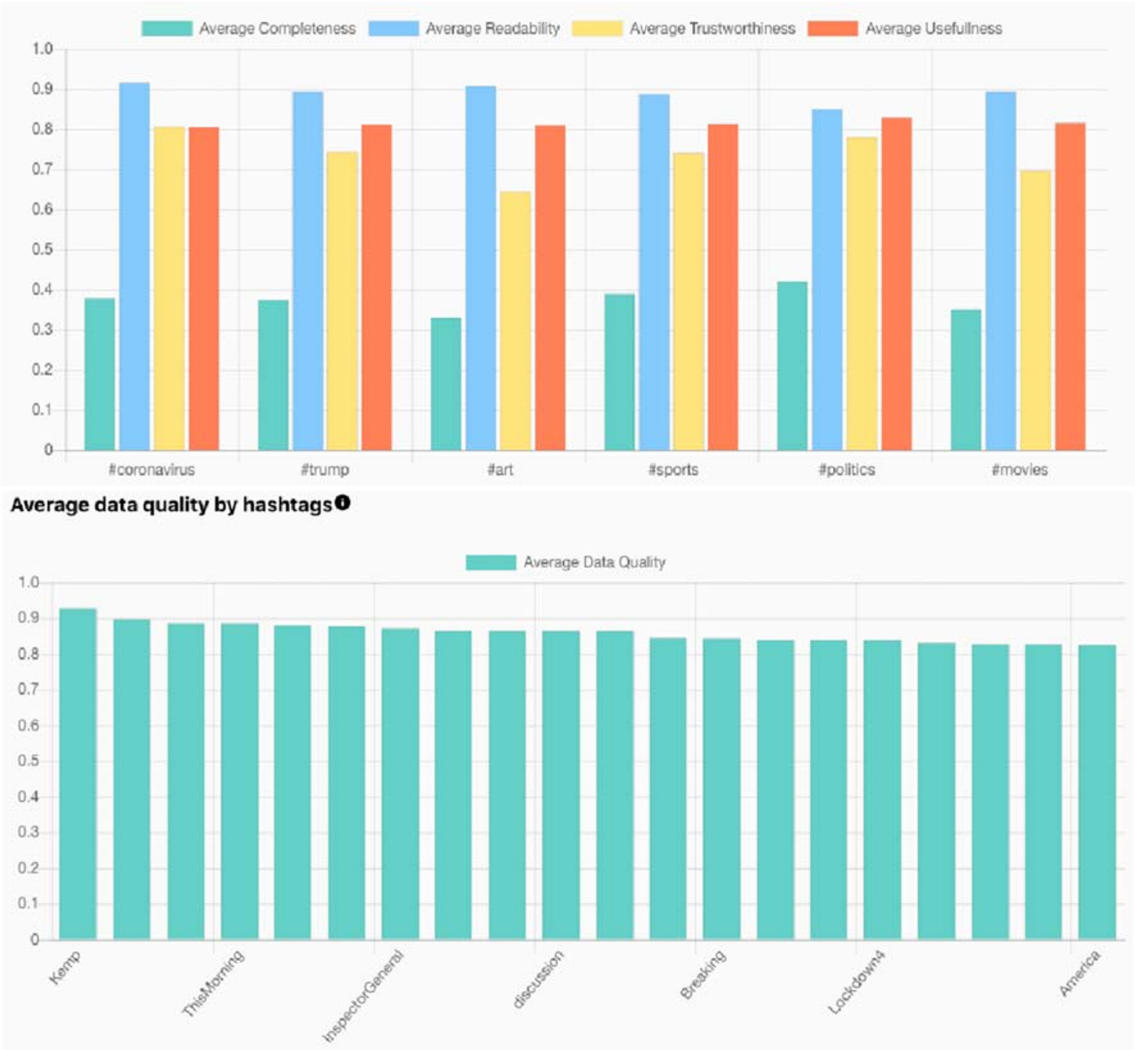
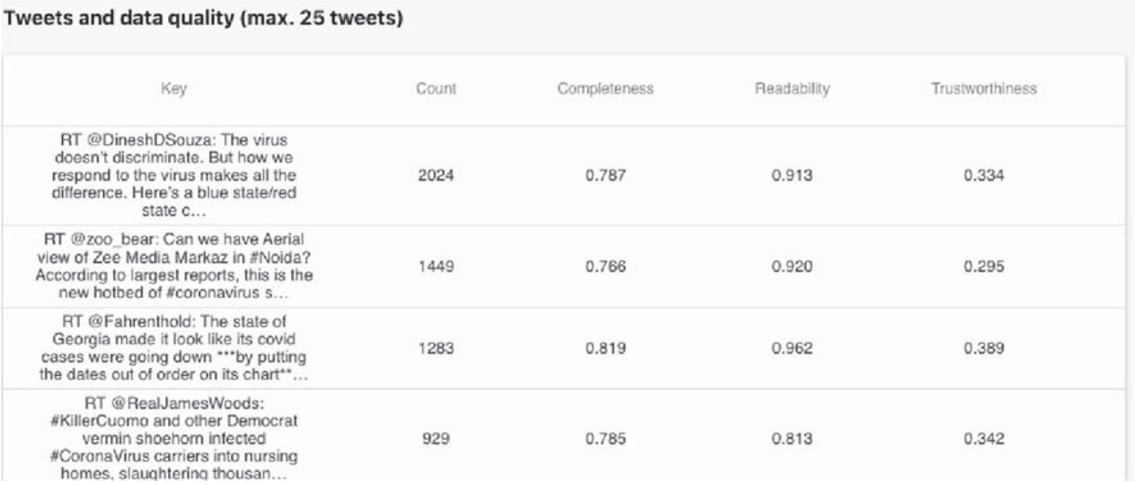


Fig. 17 DQ by keyword (top); DQ by hashtag (bottom)



Tweets and data quality (max. 25 tweets)

Key	Count	Completeness	Readability	Trustworthiness
RT @DineshDSouza: The virus doesn't discriminate. But how we respond to the virus makes all the difference. Here's a blue state/red state c…	2024	0.787	0.913	0.334
RT @zoo_bear: Can we have Aerial view of Zee Media Markaz in #Noida? According to largest reports, this is the new hotbed of #coronavirus s…	1449	0.766	0.920	0.295
RT @Fahrenthold: The state of Georgia made it look like its covid cases were going down ***by putting the dates out of order on its chart**…	1283	0.819	0.962	0.389
RT @RealJamesWoods: #KillerCuomo and other Democrat vermin shoehorn infected #CoronaVirus carriers into nursing homes, slaughtering thousan…	929	0.785	0.813	0.342

Fig. 18 DQ and text of most re-tweeted messages

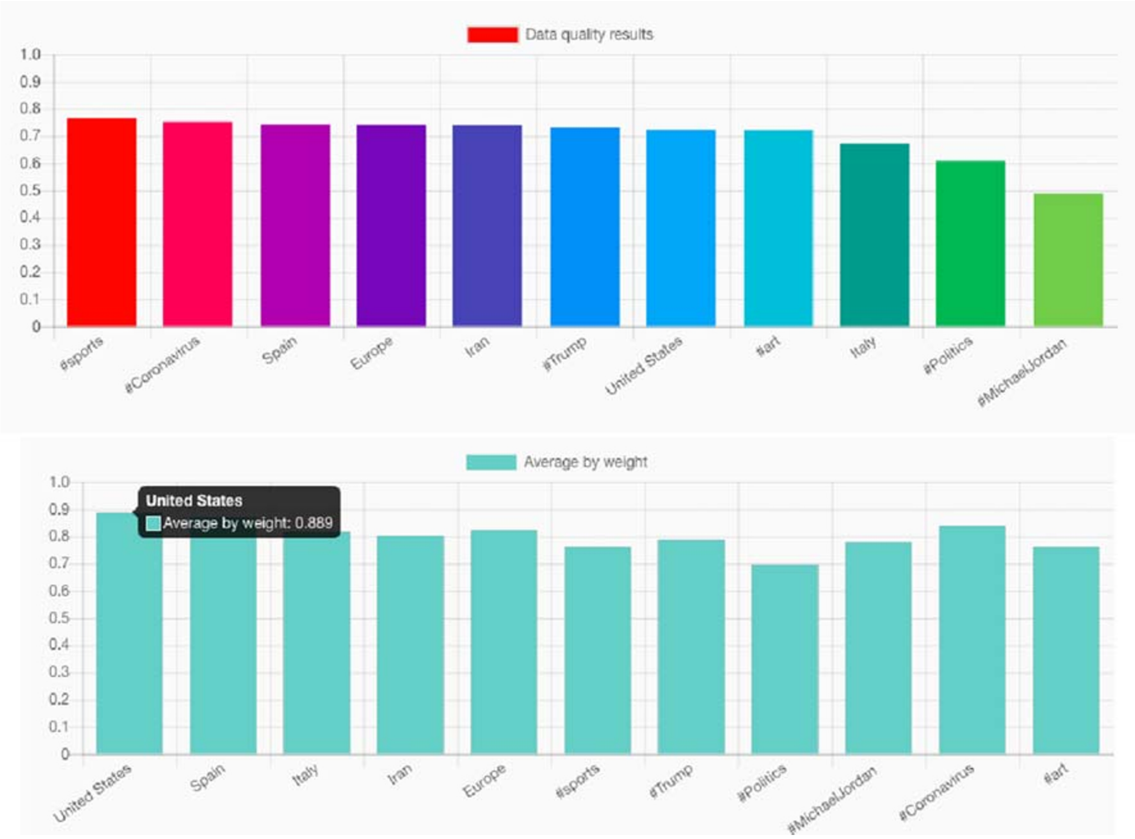


Fig. 19 DQ of different topics with even weights (top), and using context-aware weights (bottom)

dimensions were studied. The streams were classified and grouped into topics, thus allowing to analyze the quality of streams belonging to each topic. Last, but not least, and in spite of the very small percentage of the tweets that are georeferenced, a preliminary analysis was performed showing the quality of tweets with respect to the place where they were posted.

The results reported in this paper suggest that Twitter data can be reliable for helping in decision making. This follows from the fact that most of the tweets in a stream have an overall quality above 0.7. Results also showed that the quality of a tweets varies with the topic that can be associated with them. Therefore, this result can be used when some decision must be taken in a certain context related with the topics under study. Therefore, Twitter data can be a valuable source of information that can be incorporated, reliably, in the decision-making process, if needed.

There is plenty of room for further work in this field. From a technological point of view, the software tools presented here can be adapted with little extra effort, to acquire data from other social networks, or other kinds of Big Data sources. Also, new and more sophisticated visualization tools could extend and enhance the user interface. From a theoretical point of view, new DQ dimensions and metrics for this or other settings (along the lines of Firmani et al. (2015)) can be defined, since this is typical context-dependent data quality problem.

Acknowledgments Alejandro Vaisman was partially supported by the Argentinian Scientific Agency, PICT-2017 Project 1054. The authors would like to remark that an extended abstract introducing a preliminary version of the system discussed here, has been presented in the ADBIS 2018 conference (Arolfo and Vaisman 2018). This produces a small overlap with the present manuscript, mainly in the sections that introduce the background of the work, which is necessary to make the paper self-contained. The work here presents a completely new system, which has been redesigned, through a new architecture enhancing robustness and efficiency. The new system can store and retrieve past searches efficiently using ElasticSearch storage. Unlike (Arolfo and Vaisman 2018), which focuses on the system's features, the present manuscript addresses both, the new system's development and implementation, and the study of the quality of Twitter data.

References

Abedin, B., & Babar, A. (2018). Institutional vs. non-institutional use of social media during emergency response: A case of twitter in 2014 australian bush fire. *Information Systems Frontiers*, *20*(4), 729–740. https://doi.org/10.1007/s10796-017-9789-4.

Agrawal D., Bernstein P., Bertino E., Davidson S., Dayal, U. (2011). Challenges and opportunities with big data. https://docs.lib.

purdue.edu/cgi/viewcontent.cgi?referer=https://www.google.com.ar/&httpsredir=1&article=1000&context=cctech.

Arolfo, F., & Vaisman, A.A. (2018). Data quality in a big data context. In *Advances in databases and information systems - 22nd european conference, ADBIS 2018, budapest, hungary, september 2-5, 2018, proceedings, lecture notes in computer science*, (Vol. 11019 pp. 159–172). New York: Springer.

Batini, C., Rula, A., Scannapieco, M., Viscusi, G. (2015). From data quality to big data quality. *Journal of Database Management*, *26*(1), 60–82.

Batini, C., & Scannapieco, M. (2006). *Data quality: concepts, methodologies and techniques. Data-centric systems and applications*. New York: Springer.

Bolchini, C., Curino, C.A., Quintarelli, E., Schreiber, F.A., Tanca, L. (2007). A data-oriented survey of context models. *SIGMOD Record*, *36*(4), 19–26. https://doi.org/10.1145/1361348.1361353.

Byrd, K., Mansurov, A., Baysal, O. (2016). Mining twitter data for influenza detection and surveillance. In *2016 IEEE/ACM international workshop on software engineering in healthcare systems (SEHS)* (pp. 43–49), https://doi.org/10.1109/SEHS.2016.016.

Cai, L., & Zhu, Y. (2015). The challenges of data quality and data quality assessment in the big data era. Data Science Journal 14(2),https://doi.org/10.5334/dsj-2015-002.

Chang, W., & Chen, Y. (2019). Way too sentimental? a credible model for online reviews. *Information Systems Frontiers*, *21*(2), 453–468. https://doi.org/10.1007/s10796-017-9757-z.

Ciaccia, P., & Torlone, R. (2011). Modeling the propagation of user preferences. In *Proceedings of conceptual modeling – ER* (pp. 304–317). Berlin: Springer.

English-words project (2018). https://github.com/dwyl/english-words.

Firmani, D., Mecella, M., Scannapieco, M., Batini, C. (2015). On the meaningfulness of big data quality (invited paper). Data Science and Engineering pp 1–15.

Fornacciari, P., Mordonini, M., Tomaiuolo, M. (2015). Social network and sentiment analysis on twitter: towards a combined approach. In *Proceedings of the 1st international workshop on knowledge discovery on the WEB, KDWeb 2015, Cagliari, Italy, September 3-5, 2015* (pp. 53–64).

Guruprasad, H.S., Suprajha, S., Yogitha, C., J Sanghvi, A. (2015). A study on sentiment analysis using tweeter data. 1, 213–218.

Hao, M.C., Rohrdantz, C., Janetzko, H., Dayal, U., Keim, D.A., Haug, L., Hsu, M. (2011). Visual sentiment analysis on twitter data streams. In *2011 IEEE conference on visual analytics science and technology, VAST 2011, providence, rhode island, USA, October 23-28, 2011* (pp. 277–278).

Lukyanenko, R., Wiggins, A., Rosser, H.K. (2020). Citizen science: An information quality research frontier. *Information Systems Frontiers*, *22*(4), 961–983. https://doi.org/10.1007/s10796-019-09915-z.

Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for computational linguistics (ACL) system demonstrations, http://www.aclweb.org/anthology/P/P14/P14-5010* (pp. 55–60).

Marotta, A., & Vaisman, A.A. (2016). Rule-based multidimensional data quality assessment using contexts. In *18Th international conference, dawak 2016, porto, portugal, september 6-8, 2016, proceedings* (pp. 299–313).

Nodejs & express. OpenJS Foundation (2020). https://nodejs.org/docs/latest-v9.x/api/.

Poeppelmann, D., & Schultewolter, C. (2012). Towards a data quality framework for decision support in a multidimensional context. *IJBIR*, *3*(1), 17–29.

Saha, B., & Srivastava, D. (2014). Data quality: the other face of big data. In *IEEE 30th international conference on data engineering, chicago, ICDE 2014, IL, USA, March 31 - April 4, 2014* (pp. 1294–1297), https://doi.org/10.1109/ICDE.2014.6816764.

Salvatore, C., Biffignandi, S., Bianchi, A. (2020). Social media and twitter data quality for new social indicators. Social Indicators Research. https://doi.org/10.1007/s11205-020-02296-w.

Scannapieco, M., & Catarci, T. (2002). Data quality under a computer science perspective. *Archivi & Computer*, *2*, 1–15.

The Economist. Data, data everywhere (2008). https://www.economist.com/node/15557443.

The United Nations Economic Commission for Europe (UNECE)-Task Team on Big Data. Classification of types of big data (2007). https://statswiki.unece.org/display/bigdata/Classification+of+Big+Data.

Soto, A.J., Ryan, C., Silva, F.P., Das, T., Wolkowicz, J., Milios, E.E., Brooks, S. (2018). Data quality challenges in twitter content analysis for informing policy making in health care. In *51st hawaii international conference on system sciences, HICSS 2018, hilton waikoloa village, hawaii, USA, January 3-6, 2018*.

Stefanidis, K., Pitoura, E., Vassiliadis, P. (2011). Managing contextual preferences. *Information Systems*, *36*(8), 1158–1180.

Strong, D.M., Lee, Y.W., Wang, R.Y. (1997). Data quality in context. *Communications of the ACM*, *40*(5), 103–110. https://doi.org/10.1145/253769.253804.

Wagner, S., Toftegaard, T.S., Bertelsen, O.W. (2011). Increased data quality in home blood pressure monitoring through context awareness. In *5th international conference on pervasive computing technologies for healthcare, Dublin, Ireland* (pp. 234-237).

Wang, R.Y., & Strong, D.M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, *12*(4), 5–33.

Ye, H.J., Chua, C.E.H., Sun, J. (2019). Enhancing mobile data services performance via online reviews. *Information Systems Frontiers*, *21*(2), 441–452. https://doi.org/10.1007/s10796-017-9763-1.

Zadeh, A.H., Zolbanin, H.M., Sharda, R., Delen, D. (2019). Social media for nowcasting flu activity: Spatio-temporal big data analysis. *Information Systems Frontiers*, *21*(4), 743–760. https://doi.org/10.1007/s10796-018-9893-0.

Franco Arolfo received a BA in Computer Engineering from the Instituto Tecnol?gico de Buenos Aires (ITBA), Argentina, in 2018. His final project was awarded best dissertation of his promotion. He worked as a Software Engineer at MuleSoft, a San Francisco based tech startup acquired by Salesforce in 2018, focusing on the Platform-as-a-Service product of the company for 5 years. He is currently settled in The Netherlands, focusing on cloud infrastructure technologies.

Kevin Cortés Rodriguez received a BA in Computer Engineering from the Instituto Tecnol?gico de Buenos Aires (ITBA), Argentina, in 2020. He worked as a Transactional Web Services Analyst at Prisma Medios de Pago, an Argentinian leading company dedicated to developing and marketing multi-brand and multi-platform processing and payment solutions. He currently works as a Solutions Architect at Amazon Web Services within the public sector for the southern region in Latin America.

Alejandro Vaisman received a BA degree in Civil Engineering, a BA in Computer Science, and a PhD in Computer Science from the University of Buenos Aires (UBA), under the supervision of Prof. Alberto Mendelzon, from the University of Toronto, Canada. He was post-doctoral researcher and Lecturer at the University of Toronto. He was Associate Professor at UBA between 1994 and 2013, Vice-Head of the Computer Science Department at UBA, and chair of the Masters Program in Data Mining. He was a visiting researcher at the University of Toronto, Universidad Polit?cnica de Madrid, University of Hasselt, and Universidad de Chile. He is currently full professor at the Instititulo Tecnol?gico de Buenos Aires (ITBA), where he is also Director of the graduate program in Data Science. His research interests are in the field of databases, particularly in Business Intelligence, OLAP and Data Warehousing, the Semantic Web and Geographic Information Systems. He has authored and co-authored over 100 scientific papers presented at major database conferences and journals.