Quality of Information as an indicator of Trust in the Internet of Things

Hamza Baqa[1,3], Nguyen Binh Truong[2], Noel Crespi[3], Gyu Myoung Lee[2], Franck Le Gall[1]

[1] Easy Global Market
Sophia Antipolis, France
(hamza.baqa|franck.le-gall)@eglobalmark.com

[2]Department of Computer Science,
Liverpool John Moores University
Liverpool, United Kingdom
(n.b.truong@2015.|g.m.lee@)ljmu.ac.uk

[3]Télécom SudParis, TSP
Paris, France
(hamza.baqa|noel.crespi)@telecom-sudparis.eu

***Abstract*— The past decade has seen a rise in complexity and scale of software systems, particularly with the emerging of the Internet of Thing consisting of large scale and heterogeneous entities which results in difficulties in providing trustworthy services. To overcome such challenges, providing high quality information for IoT service provider as well as monitoring trust relationship of end-users toward the services are paramount. Such trust relationships are user-oriented and subjective phenomenon that hook on specific quality of data. Following this catalyst, we propose a mechanism to evaluate quality of information (QoI) for streaming data from sensor device; then use the QoI evaluation score as an indicator of trust. Concepts and assessment methodology for QoI along with a trust monitoring system are described. We also develop a framework that classifies streaming of data based on semantic context and generate QoI score as a relevant input for a trust monitoring component. This framework enables a dynamic trust management in the context of IoT for both end-users and services that empowers service providers to deliver trustworthy and high quality IoT services. Challenges encountered during implementation and contribution in standardization are discussed. We believe this paper offers better understanding on QoI and its importance in trust evaluation in IoT applications; also provides detailed implementation of the QoI and Trust components for real-world applications and services.**

Keywords—Quality of Information; Semantics; Trust; Knowledge; Reputation; Experience; Linked Data.

I. Introduction

The past decade has seen a dramatic rise in complexity and scale of software systems, especially due to the introduction of the Internet of Things (IoT). Companies and innovators have started to build, deploy, and incorporate IoT systems over large geographical areas consisting variety of technologies from data collection to smart decision making [1]. The large scale brings complexity: each local site or organisation makes an individual choice of technologies and deploys an interdependent stack of devices, networks, and services that are offered for exploitation by application developers. To reach the full potential of such IoT systems, ensuring the high quality of collected data and providing trustworthy services is a must. However, the diversity and huge number of involved entities, and wide geographical distribution of contexts in IoT systems make it challenged to completely satisfy such requirements. A traditional approach is directly managing “things” by using an identification method that things should be first identified in order to be integrated into an IoT network to provide data and get control. Most of identification technologies consist of a naming, addressing or tagging phase as a prerequisite, either during manufacturing like attributing a MAC address for classical network devices, embedding a SIM card in a telecom style, or giving attached tag later like Barcode, RFID tag. Unfortunately this tagging process can be very costly and tedious since the number of "things" is potentially in the order of trillion [2].

In this perspective, we proposes perceived information quality based on semantic context to evaluate as a relevant factor of perceived risk and trusting beliefs [3]. In the same sense, authors in [4] show how it is possible to align credibility, an extrinsic data quality (DQ) with trust. Our previous paper also pointed out the need for assessing DQ provided by mobile device owners in the evaluation of trust in the crowdsourcing context [5]. These research works serve as an alignment for the investigation on a correlation between trust and intrinsic data qualities. Following this catalyst, in this paper we propose a framework for the trust management based on the quality of information (QoI) scores for the QoI dimensions: syntactic accuracy, semantic accuracy, completeness, uniqueness, and timeliness. We thereby understand information quality assessment as the "process of assigning numerical and categorical values to QoI dimensions" [6, 7].

Our methodology is to use these dimensions to generate a concrete score based on a specific knowledge, a referent ontology, which provides a meaningful input for a trust monitoring system that evaluates trust relationships between users and service providers. We consider the QoI as referring to a degree of a dataset that fits or fulfils a form of usage, relating the DQ concerns to a specific use-case. This paper focuses on the development and evaluation of a QoI module. We primarily followed the QoI definition in [7, 8, 9] to classify the QoI rules. Then, we propose a way to generate a score for these dimensions. We finally evaluated our results by utilising these scores as an important indicator of trust called Knowledge in the Reputation-Experience-Knowledge (REK) trust model proposed in [10, 5] for evaluating trustworthiness of IoT services and applications.

With the contribution of this paper, we hope to initiate a community-wide process of further enhancing and complementing the quality rules in assessing QoI [11, 12]. Such a community effort could homogenize the interfaces for monitors and probes and increase our ability to explore new combinations and improve the DQ evaluation. Moreover, we hope to encourage the use of a trust monitoring system for delivering better quality of IoT applications and services.

The paper is organized as follows. Section II describes the background and related work. Section III describes the development of the quality information module. Section IV describes the trust evaluation module. Section V describes the implementation and exemplary use of the QoI and the Trust monitor. Section VI describes the results of a preliminary evaluation. Section VII discusses the obtained results and Section VIII summarizes and concludes the paper.

II. BACKGROUND AND RELATED WORK

A. DQ Assessment

The recent years have seen an important shift of interest in how the decision-makers deal with the quality of data. It allows the use of data more efficiently and effectively [13, 14]. For example, decision makers need to utilize soft data, a subjective assessment or future trend forecast which can be used for decision making, such as the marketing strategies of competitors in order to change or adapt the marketing strategy of the company accordingly [14]. There is no agreement on a standard definition of DQ that can be applied across all data domains [15]. The intended use is commonly described as a multi-dimensional concept consisting of a set of quality attributes, called DQ dimensions which are determined by the data users [16, 17]. In this study, it is assumed that information to be of high quality when they are “fit for use by data consumers”, and they end up by selecting 15 different dimensions and grouped them under four different categories such as Intrinsic, Accessibility, Contextual, and Representational as depicted in Table 1.

TABLE 1. DQ DIMENSIONS PROPOSED BY WANG AND STRONG

DQ category	DQ dimensions
Intrinsic DQ	Accuracy, Objectivity, Believability, Reputation
Accessibility DQ	Accessibility, Access security
Contextual DQ	Relevancy, Value-added, Timeliness, Completeness, Amount of data
Representational DQ	Interpretability, Ease of understanding, Concise representation, Consistent representation

To facilitate a structured acquisition of quality requirements, our reflexion are based on the papers [13, 18, 19] which covered 11 experts: 5 from the academic sector and 6 from the national governmental entities, 3 on whom were NSO representatives. The set of DQ dimensions has been tested with experts using four different data usage contexts: data for scientific research, data for decision-making, data for analysis the progress of research object during the reporting period, data for research object modelling and forecasting. The output of survey propose 13 dimensions for the quality assessment. And as far we are using the semantical data, we can abstract the dimensions related to a specific context (decision-making, scientific research). We end up then by 5 main dimensions shown in Table 2: 1) Semantical Accuracy, 2) Syntactical Accuracy, 3) Completeness, 4) Timeliness and 5) Uniqueness.

TABLE 2. DQ DIMENSIONS PROPOSED BY WANG .

DQ dimensions	Explanation
Semantical Accuracy	Describes the proximity of data value representations of an object related to their real-world states
Syntactical Accuracy	A value is syntactically accurate, when it is part of a legal value set for the represented domain or it does not violate syntactical rules defined for the domain
Completeness	Which is characterized in terms of the presence/absence of values
Timeliness	Which aims to identify the current values of entities represented by tuples in a (possibly stale) database and to answer queries with the current values;
Uniqueness	The degree to which data is free of redundancies in breadth, depth, and scope

In this paper, these DQ dimensions are used as a metric to judge the impact on DQ. The authors in [9, 19] showed that any QoI problem can be expressed as difficulties at the level of these dimensions.

B. DQ in Semantic Web

DQ should be defined in the context of being fit for a particular use. Data ‘fitness for use’ depends on the application of the data, the characteristics of quality that are necessary for that specific purpose and on the user’s expectations of what they define to be useful information [13]. In this perspective, it is more interesting to check the DQ from a semantical level. Semantic data refers to data whose meaning has been made explicit in the form of meta-data. Such metadata may then be used in semantics-based approaches to manage the data. The perhaps most prevalent approach to represent semantic data and its meta-data is based on the Resource Description Framework (RDF) represented by triples, which allow the definition of statements in a subject, predicate, object format. This combination of two entities (subject, object) and a relationship (predicate) is called a triple. Thus, with RDF or OWL (Web Ontology Language) it is possible to enrich the data and define relationships between the different things. For example, in IoT domain, we can provide enhanced meaning for sensor observations so as to enable situation awareness. It enhances meaning by adding semantic annotations to existing standard sensor languages of the SWE. These annotations provide more meaningful descriptions and enhanced access to sensor data than SWE alone, and they act as a linking mechanism to bridge the gap between the primarily syntactic XML-based metadata standards of the SWE and the RDF/OWL-based metadata standards of the Semantic Web [14].

C. Semantic Validator

Semantic ontology validator is a web application which integrates the ontology and data validation functionalities in a web-based client-server architecture [15]. It detects syntactic and semantic issues if any, and produces a detailed test report at the end of the process. According to the predefined reference ontology, the “fitness for use’’ knowledge, it checks the syntactic errors (syntactic accuracy) based on a configuration file, where we put all the configurations related to the data quality rules. A reasoner is also used to enable the logical level verification of the RDF description such as the respect of subsumption relationships between classes, restrictions on class properties and cardinality, etc. [18]. There are three main functionalities in the main system, namely XML Parser or JSON Parser according to the format of the input file, RDF Parser, and Validation as illustrated in Fig. 1.

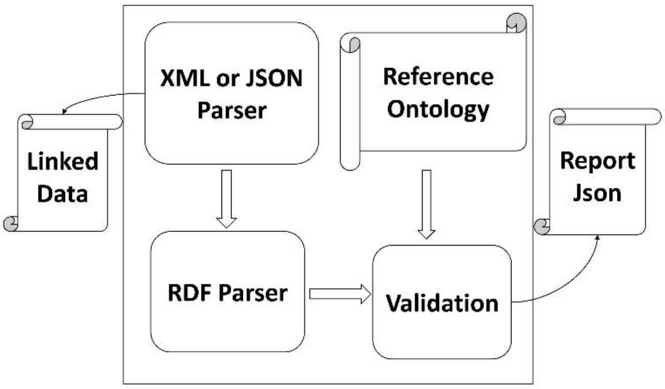

Fig. 1: Ontology web app architecture

- **Syntax Parser:**

It is indicated as XML or JSON Parser in Fig. 1 which also supports other formats such as RDF/XML, OWL/XML, and JSON-LD initially performing syntax check of linked data. If any error occurred, the validation process will break. If input files are in incorrect format, it will directly passed to the RDF parser module.

- **RDF Parser**:

This module takes a document either a validated XML, a JSON or another supported format file and verifies if the document represents a valid RDF model (ref: https://www.w3.org/TR/2004/REC-rdf-concepts-20040210). If it respects the specification of the RDF model, triples in this model are extracted to serve as the input for the next validation step which is the "validation" module. Some errors are detectable in this step, for example the "class not declared" problem when a class is not defined in the current document nor has a prefix. However as these errors do not prevent the extraction of triples, the validation process will be processed to the next step.

- **Validation:**

This module takes the reference ontology, the triples from the previous step and the configuration file as input. The configuration file lists the different tests and quality rules to be performed called DQ inspectors. A reasoner is also used to enable the logical level verification of the RDF document such as the respect of relationships between classes, restrictions on class properties and cardinality, etc.

D. DQ in IoT System

DQ is crucial to gain user engagement and acceptance of the IoT paradigm and services. The connected environment makes the DQ a major concern in providing IoT services in the sense that the IoT data serves as a base to extract insights about people, entities and phenomenon. In fact, data trust is crucial for the user engagement and acceptance of the IoT paradigm [19]. Data suffers from quality problems will fail to represent the reality. The decisional and operational levels of any business or organization can be rolled-back [11].

A wide range of IoT applications have been developed and deployed in industries such as in healthcare [20], traffic management [21], or smart parking [22]. With the increasing amount of applications deployed over the same IoT systems, it became critical to monitor quality and usage of the systems and to evaluate whether the technical capabilities are well aligned with the needs of application developers, users, and other stakeholders [23].

IoT systems are designed in a multi-layered architecture including sensing, networking, and application layers [24].

- The application layer manages the systems' functionalities and exporting all to the final users to address their needs and expectations.
- A service layer enables the application by managing the interactions between the applications and the IoT systems. Commonly, these services expose APIs that offer access to data, streams, and actuator capabilities [25].
- The networking layer supports data transfer over a wired or wireless network between the sensing and application layers, which may use private and public clouds infra-structures [26].
- The sensing layer contains hardware such as sensors, actuators, and RFID to acquire data for monitoring business goals and exert control over the physical world.

Such architecture facilitates testing versions of components, subsystems, and applications from a multitude of vendors to determine user preferences and needs. For example, probes may be injected into the service layer for gathering information provided by the IoT system about the environment and for obtaining cues about the quality of that information. Probes may also be injected into the application layer for collecting data about application usage and quality and for interacting with the users, e.g. with in-product surveys. If these two perspectives could be related, innovation experiments could be performed that try to put dependent variables, e.g. the users' trust, into a relationship with independent variables, e.g. the quality of IoT data. Hence, the aforementioned problems that threatens the quality of produced data occur in different layers of the IoT system model.

III. QoI Assessment Methodology

A. QoI Module

The QoI module classifies DQ problems and calculates QoI scores for the QoI dimensions: syntactic accuracy, semantic accuracy, completeness, uniqueness, and timeliness based on previously defined DQ rules. The information quality score metrics are based on the simple ratio calculation as described in [27]. The simple ratio is measured by subtracting the ratio between the total numbers of axioms that violate a DQ rules for a dimension and the total number of axioms (DV), we proceed with the same manner for each dimension to generate a score for each dimension QoI_{dim} in (1). For the final score, we choose an “importance weight” approach[1], which is intended for programmers, not data analysts" to influence the final score w_{dim} (2). This weight can be attributed directly by the user or in the semantic description using Weighting Ontology[2] *(wo:weight).*

$$QoI_{dim} = 1 - (\frac{\mathrm{DV}}{\mathrm{T}}) \quad (1)$$

$$QoI = \sum_{i} w_i \times QoI_i \quad (2)$$

where *i* represents the dimensions. The QoI supports also the case where the user provides a weight for each property for more precision w_p (3), the same as the first case, the user still can manually provide it or directly to the annotation (*wo:weight* as a subclass of the main *wo:weight*) according to the Weighting Ontology.

$$QoI = \sum_{i} w_{i,j} \times w_i \times QoI_i \quad (3)$$

where $w_{i,j}$ is the weight of property *j* of dimension *i*, and w_i is the weight of dimension *i.*

B. DQ Rules and Dimensions

In the following sections, we explain the semantics and composition of each dimensions based of the quality rules that provides the semantic validator (Section II-C) and to their definitions in Section II-A and Section II-B. Table 3 shows the mapping of the quality inspectors, plug-ins that can be customised by the user, implemented in the ontology validator and the QoI dimensions:

TABLE 3. DQ DIMENSIONS WITH DQ RULES

DQ dimensions	**DQ inspectors**
Semantical Accuracy	The resonning capabilities [7]
Syntactical Accuracy	Literal Inspector: Checks literals for syntactically correct language codes, syntactically correct datatype URIs (using the same rules as the URIInspector), and conformance of the lexical form of typed literals to their datatype.
Completeness	ConsistentType Inspector: checks that every subject in the model can be given a type which is the intersection of the subclasses of all its "attached" types -- a "consistent type". For example, if the model contains three types Top, Left, and Right, with Left and Right both being subtypes of Top and with no other subclass statements, then some S with rdf:types Left and Right would generate this warning. VocabularyInspector: checks that every URI in the model with a namespace which is mentioned in some schema is one of the URIs declared for that namespace -- that is, it assumes that the schemas define a closed set of URIs.
Timeliness	Time Inspector: identify instances that represent an outdated state of the corresponding real world entity.
Uniqueness	PropertyInspector : checks that every predicate that appears in the model is declared in some -assumed schema or owl:imported model -- that is, is given rdf:type rdf:Property or some subclass of it.
	ClassInspector: Checks that every resource in the model that is used as a class, ie that appears as the object of an rdf:type, rdfs:domain, or rdfs:range statement, or as the subject or object of an rdfs:subClassOf statement, has been declared as a Class in the -assumed schemas or in the model under test.

IV. TRUST EVALUATION BASED ON QOI

This section introduces the necessity of QoI as an indicator of trust in variety of IoT applications and services. An evaluation model based on QoI and users’ feedback is also presented.

A. QoI as an indicator of Trust

Trust can be roughly defined as belief of a trustor in a trustee that the trustee will accomplish a given task satisfying trustor’s expectation. Evaluation of trust could support a trustor to lower vulnerabilities and potential risks as well as to overcome perception of uncertainty when making any decision [5].

Various use-cases have been investigated in which trust is utilized for supporting users to select proper options in a recommendation system and deliver better quality of services (QoS). The need for trust in IoT applications and services can be illustrated by taking the smart parking use-case in our ongoing Wise-IoT[3] project as an example. In this smart parking service, an end-user requests an available parking lot which is close to a destination in a specific time slot. Under the context of trust, the user expects to find a parking lot that she trusts to park her car. Therefore, parking sensors, which are used to indicate the availability of a parking lot, and traffic sensors, which are used to estimate the estimated time arrival (ETA) from user’s position to the parking place, should be working correctly. Therefore, the evaluation of QoI of such sensors is an important aspect to indicate the status of the parking sensors and the traffic sensors. However, only QoI scores might not be enough for illustrating the users’ trust toward a parking lot. Other factors also contribute to how a user selects a parking lot including user’s preferences, previous experience, or the reputation of the parking service. Such factors could be expressed and quantified by assembling users’ experiences and opinions using a feedback mechanism. Nevertheless, as any IoT applications and services heavily depends upon collected data, QoI plays a crucial role in indicating and evaluating trust between users and IoT services.

B. Utilization of the REK Trust Model based on QoI and Feedback

To establish and evaluate trust relationships between service requesters (i.e., trustors) and service providers (i.e., trustees), we leverage the REK trust model in the IoT proposed in [5, 10], which consists of the three major trust indicators, namely Reputation, Experience and Knowledge. Knowledge indicator is as “*direct trust*” inferred from attributes of a trustee whereas Experience and Reputation are as “*indirect trust*” calculated from previous interactions illustrating personal opinion and global perspective toward the trustee, respectively. In this paper, the REK model is employed, taking the evaluation of QoI and user’s feedback into account, as illustrated in Fig. 2. Knowledge indicator is evaluated as QoI score; other trust-related attributes

[1] https://www.stata.com/statalist/archive/2003-02/msg00525.html

[2] http://smiy.sourceforge.net/wo/spec/weightingontology.html

[3] http://wise-iot.eu/en/home/

are neglected due to unavailability or not being suitable to collect; however, in some use-cases, some other useful information could be gathered and plays as supplemental factor in evaluating Knowledge. The two indicators Reputation and Experience are calculated based on previous research works [5, 10] and briefly presented below.

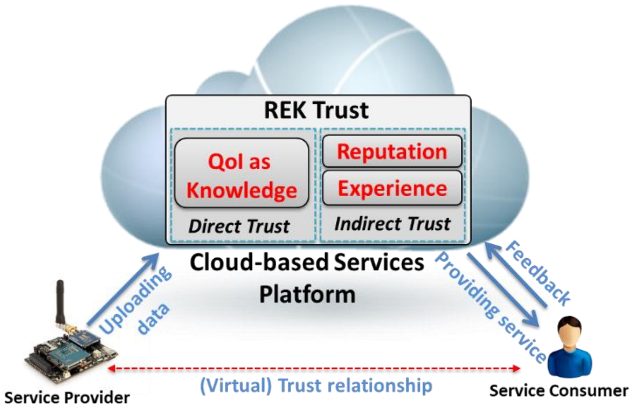

Fig. 2. Utilization of the REK Trust Model based on QoI and Feedback in variety of IoT applications and services

1) Experience Indicator Calculation

Experience indicator is calculated based on users' feedback information in which Experience value increases due to positive feedbacks and decreases due to negative feedback. Experience value also decays if there are no interactions after a period of time. The amount of the increase, decrease and decay depends on the intensity of interactions, feedback values, and the current Experience value that are analysed and modelled using difference equations proposed in [5, 10]. Let denote Exp_t is the Experience value at the time *t* (between any two users), $init_{Exp}$ is the initial *Exp*, max_{Exp} is the maximum *Exp* ($\alpha < max_{Exp}$); and α is the *maximum increase value* of *Exp*. We apply those mathematical models to calculate Experience based on feedback as following:

- **Increase due to positive feedback** $feedback > \theta_{positive}$

$$Exp_{t+1} = Exp_t + feedback_t \times \Delta Exp_{t+1} \qquad (4)$$

$$\text{where } \Delta Exp_{t+1} = \alpha \times (1 - \frac{Exp_t}{max_{Exp}}) \qquad (5)$$

- **Decrease due to negative feedback** $feedback > \theta_{negative}$

$$Exp_{t+1} = Max\langle min_{Exp}, Exp_t - (1 - feedback_t) \times \beta \times \Delta Exp_{t+1}\rangle \qquad (6)$$

in which β is a parameter showing the *rate of decrease*.

- **Decay due to neutral feedback or no interactions**

$$Exp_{t+1} = Max\langle init_{Exp}, Exp_t - \Delta decay_{t+1}\rangle \qquad (7)$$

$$\text{where } \Delta decay_{t+1} = \delta \times \left(1 + \gamma - \frac{Exp_{t-1}}{max_{Exp}}\right) \qquad (8)$$

in which δ is *minimal decay value* and γ is *decay rate*.

2) Reputation Indicator Calculation

Reputation indicator can be calculated using a graph analysis algorithm on the Experience topology between users which is similar to Google PageRank [28] and weighted PageRank [29]. The modified weighted PageRank algorithm for Reputation indicator under the context of trust have been proposed in [10, 5]. We have apply this model in some real-world IoT applications considering Experience indicator (i.e., $Exp(i, A)$ between *i* and *A*) as following:

- **Positive Reputation**: based on positive feedback

$$Rep_{Pos}(A) = (1 - d) + \mathrm{d} \times \left(\sum_{\forall i} Rep_{Pos}(i) \times \frac{Exp(i, A)}{C_{Pos}(i)}\right) \qquad (9)$$

Where: $C_{Pos}(i) = \sum_{Exp(i,j)>threshold} Exp(i, j)$.

- **Negative Reputation**: based on negative feedback

$$Rep_{Neg}(A) = (1 - d) + \mathrm{d} \times \left(\sum_{\forall i} Rep_{Neg}(i) \times \frac{1 - Exp(i, A)}{C_{Neg}(i)}\right) (10)$$

Where: $C_{Neg}(i) = \sum_{Exp(i,j)<threshold} (1 - Exp(i, j))$.

- **Overall Reputation**: combines two positive and negative reputations

$$Rep(A) = \max\left(0, Rep_{Pos}(A) - Rep_{Neg}(A)\right) \qquad (11)$$

in which *d* is the damping factor. Detailed theoretical model, analysis and implementation mechanism can be found in our research articles [5, 10].

V. IMPLEMENTATION

A. QoI-based Trust Monitoring System Architecture

Trust monitoring system can be used in various IoT services and applications for supporting recommendation and decision making. The conceptual trust-based system architecture along with basic components are illustrated in Fig. 3. Here, the Trust Monitor basically takes QoI information of traffic sensors and parking sensors from the QoI monitor and user feedback from the feedback mechanism in the Adherence Monitor as its inputs for trust evaluation process. Also, the Trust Monitor provides an API to the IoT Recommender for enquiring about trust evaluation value between users and service providers.

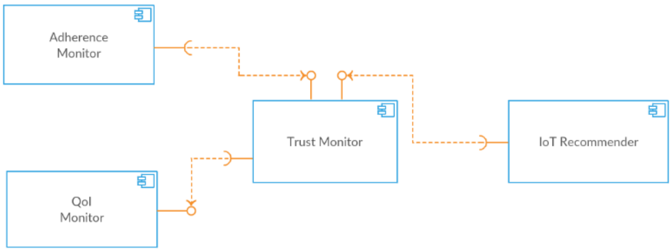

Fig. 3: UML component diagram for the trust-based system

All of the components in the system are deployed using RESTful Web-service API for being easy to implement and interact, maintainable, extensible, and scalable. Detailed implementation for the QoI monitor and Trust Monitor are then described below.

B. QoI Monitor Implementation

Based on the previous section, the architecture of the QoI module is composed by two basic layer: i) the semantic validator web service that we have developed (see section II-C) and ii) a calculation module that takes the output of the first module and generate a score. The calculation module support itself two different configurations. Fig. 4 shows the case where the user provide directly the weights to the scoring module.

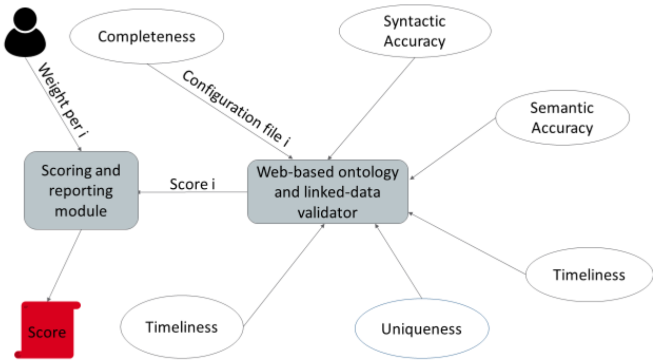

Fig. 4: User as a weight provider

The second case is to provide the weighing factors within the annotations (*wo:weight*), for example, we can assign integer values ranging from one meaning "*slightly important*" to five meaning "*task critical*", the Weighting Ontology define a vocabulary for that purpose (*wo:max_weight* which is a decimal that describes the maximum, in our case the "*important*" task and *wo:min_weight* for the "*less important*").

C. Experience and Reputation Indicators Implementation

- Experience Parameters Setting

In real implementation, feedback value obtained from users is in form of ratings from one star (*) to five stars (*****) then the value is normalized to the range (0, 1). Experience values are also normalized to the range (0, 1) by setting up $min_{Exp} = 0$ and $max_{Exp} = 1$. Below is the detailed parameters using in our implementation.

TABLE 4. PARAMETERS SETTINGS FOR THE SIMULATION OF EXPERIENCE

Parameters	Values	Parameters	Values
max_{Exp}	1	γ	0.005
min_{Exp}	0	δ	0.005
$init_{Exp}$	0.3	$\theta_{negative}$	0.3
α	0.1	$\theta_{positive}$	0.6
β	2		

- Reputation Parameters Setting

We use the iterative method described in [30] for solving the matrix equation of the reputation model with the damping factor d = 0.85 same as in Google PageRank [28]. In the smart parking use-case, we consider some scenarios in which the number of users with traffic sensors and parking sensors (i.e., network size) as *N*=400, 800 and 1600 with the tolerance = 10^{-3} which is suitable for ranking *N* entities. The tolerance is *2-norm vector* of the *Rep* vectors in two consecutive iterations.

D. Trust Calculation

Trust value is an aggregation of the three indicators Knowledge, Experience and Reputation. Although there are variety of techniques to combine the three trust indicators such as Bayesian neutron networks, fuzzy logic and machine learning depending on specific context, here a simple weighted sum for trust value between trustor A and trustee B is used as follows:

$$Trust(A,B) = \alpha Rep(B) + \beta Exp(A,B) + \gamma QoI(B) \qquad (9)$$

in which $\alpha, \beta, \gamma > 0$ satisfying $\alpha + \beta + \gamma = 1$. For simplicity, we set $\alpha = \beta = \gamma = 1/3$. These weighting factors can be autonomously tuned using machine learning for analysing feedback from users. Detailed implementation and source code of the trust monitor including Experience and Reputation indicators calculation mechanisms can be found here[4].

VI. PRELIMINARY EVALUATION AND DISCUSSION

A. Challenges in Real-world Applications and Service Deployment

- **Interoperability within the other components**

Interoperability is one of the major challenges, particularly within component based software development environments, an approach in which prefabricated reusable software components from independent sources are assembled together to build applications. There are many aspects related to component interoperability, including syntactic agreements on method names, behavioural specifications of components, service access protocols, business domain knowledge, negotiation of Quality of Service and other non-functional properties.

- **Building real-time system**

One other big challenge that we faced during the implementation phase is to build a component that generate an output for each semantic description in real-time, especially in smart city context that synchronize data across thousands or even millions IoT devices, which makes a real-time aspect very challenging.

- **Efficient Trust Evaluation mechanisms**

Besides the calculation of the QoI for data streaming, the calculation of the REK trust evaluation mechanism should be taken into account. Unlike the Experience calculation model which is quite simple and the computation complexity is insignificant, the calculation of Reputation requires more computational resources. Based on the computational model, Reputation model can be calculated either algebraically or iteratively. The algebra traditional method to solve the matrix equations (9) and (10) takes roughly N^3 operations that is not suitable for a huge number of nodes (users). The iterative methods is much faster because the Rep_{Pos} and Rep_{Neg}vectors converge after a number of iterations as the scaling factor in terms of number of nodes is roughly linear in *logN*. Therefore, the reputation calculation mechanism is suitable to implement in

[4] https://github.com/nguyentb/TrustEvaluation

any IoT systems. However, it is not necessarily to execute the reputation calculation every time, instead it can be periodically conducted, same as Google is currently doing in their Website PageRank mechanism.

B. Contribution on Standarization

As ITU-T has newly created new Focus Groups on data processing and management (FG-DPM) as well as Data Trust group, we expect that our proposals can significantly contribute to further stimulate standardization activities in the future, taking into account QoI and trust challenges in the IoT.

VII. Conclusion and Future Work

To reach full the potential of the IoT provisions, we have investigated the DQ, QoI and the relations with trust in order for evaluating trust between end-users and service providers and empowering trustworthy IoT applications. This paper introduces a framework leveraging the trust monitoring system in relation to objective measurements of QoI DQ; as well as shows how QoI assessment and Trust evaluation are deployed in a real-world large-scale IoT environment.

The validation of the framework is under investigation. A system called self-adaptive recommender (SAR) that have being developed and deployed in our testbeds for the Wise-IoT project will be used to verify and validate the work. The SAR system offers the dynamism needed to setup experiments and harvest data streaming needed for analysing the outcomes of the framework.

Regarding to research work, the paper opens several future directions including the usage of QoI assessment and trust in variety of IoT services and the improvement of the trust evaluation model such as the integration of useful information, besides QoI, for quantifying trust and the efficiency of the reputation mechanism.

Regarding to practical deployment, we have already broken down some challenges during the implementation and deployment of the trust monitoring system framework which impose some potential research directions for dealing with.

Acknowledgment

This research was supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 723156, the Swiss State Secretariat for Education, Research and Innovation (SERI) and the South-Korean Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MISP) (No. R7115-16-0002).

References

[1] A. Sheth, P. Anantharam and C. Henson, "Physical-cyber-social computing: An early 21st century approach," *IEEE Intelligent Systems,* vol. 28, pp. 78-82, 2013.

[2] D. Uckelmann, Quantifying the value of RFID and the EPCglobal Architecture Framework in Logistics, Springer Science & Business Media, 2012.

[3] A. Nicolaou and D. McKnight, "Perceived information quality in data exchanges: Effects on risk, trust, and intention to use," *Information systems research,* vol. 4, no. 17, pp. 332-351, 2006.

[4] D. Ceolin, V. Maccatrozzo, L. Aroyo and T. De-Nies, "Linking Trust to Data Quality," in *4th International Workshop on Methods for Establishing Trust of (Open) Data.,* VU University Amsterdam, 2015.

[5] N. Truong, H. Lee, B. Askwith and G. Lee, "Toward a Trust Evaluation Mechanism in the Social Internet of Things," *SENSORS,* vol. 17, no. 6, 2017.

[6] Y. Lee, M. Strong, K. Kahn and R. Wang, "AIMQ: a methodology for information quality assessment," *Information & management,* vol. 2, no. 40, pp. 133-146, 2002.

[7] M. Ge and M. Helfert, "Data and information quality assessment in information manufacturing system," in *International Conference on Business Information Systems*, 2008.

[8] N. Askham, D. Cook, M. Doyle, H. Fereday, M. Gibson, U. Landbeck, R. .. Lee, C. Maynard, G. Palmer and J. Schwarzenbach, "The six primary dimensions for data quality assessment," DAMA UK Working Group, United Kingdom, 2013.

[9] N. Laranjeiro, S. Soydemir and J. Bernardino, "A survey on data quality: classifying poor data," in *IEEE 21st Pacific Rim International Symposium on Dependable Computing (PRDC),* 2015 .

[10] N. B. Truong, T. Um, B. Zhou and G. M. Lee, "From personal experience to global reputation for trust evaluation in the social internet of things," in *IEEE Global Communications Conference (GLOBECOM),* Singapore, 2017.

[11] C. Batini, C. Cappiello, C. Francalanci and A. Maurino, "Methodologies for data quality assessment and improvement," *ACM computing surveys (CSUR),* vol. 3, no. 41, 2009.

[12] M. H. T. Pham, "A review of quality frameworks in information systems," in *International Conference on Information Systems Technology and its Applications ISTA'2007*, 2017.

[13] S. Jesiļevska, "Data Quality Dimensions to Ensure Optimal Data Quality," *Romanian Economic Journal,* vol. 20, no. 63, 2017.

[14] A. Sheth, C. Henson and S. Sahoo, "Semantic sensor web," *IEEE Internet computing,* vol. 4, no. 12, 2008.

[15] H. Kim, A. Ahmad, J. Hwang, H. Baqa, F. LeGall, M. Ortega and J. Song, "IoT-TaaS: Towards a Prospective IoT Testing Framework," *IEEE Access,* 2018.

[16] H. Moges, V. Vlasselaer, W. Lemahieu and B.Baesens, "Determining the use of data quality metadata (DQM) for decision making purposes and its impact on decision outcomes—An exploratory study," *Decision Support Systems,* vol. 83, pp. 32-46, 2016.

[17] B. Klein and D. Rossin, "Data quality in neural network models: effect of error rate and magnitude of error on predictive accuracy," *Omega,* vol. 5, no. 27, pp. 569-582, 1999.

[18] A. Maarala, X. Su and J. Riekki, "Semantic reasoning for context-aware Internet of Things applications," *IEEE Internet of Things Journal,* vol. 2, no. 4, pp. 461-473, 2017.

[19] A. Karkouch, H. Mousannif, H. Moatassime and T. Noel, "Data quality in internet of things: A state-of-the-art survey," *Journal of Network and Computer Applications,* no. 73, pp. 57-81, 2016.

[20] M. Domingo, "An overview of the Internet of Things for people with disabilities," *Journal of Network and Computer Applications*, vol. 2, no. 35, pp. 584-596, 2012.

[21] L. Foschini, T. Taleb, A. Corradi and D. Bottazzi, "M2M-based metropolitan platform for IMS-enabled road traffic management in IoT," *IEEE Communications Magazine,* vol. 11, no. 49, 2011.

[22] Z. Ji, I. Ganchev, M. O'Droma, L. Zhao and X. Zhang, "A cloud-based car parking middleware for IoT-based smart cities: Design and implementation," *Sensors,* vol. 12, no. 14, pp. 22372-22393, 2014.

[23] K. Bratanis, D. Dranidis and A. Simons, "An extensible architecture for run-time monitoring of conversational web services," in *3rd International Workshop on Monitoring, Adaptation and Beyond*, 2010.

[24] L. Atzori, A. Iera and G. Morabito, "The internet of things: A survey," *Computer networks,* vol. 15, no. 54, pp. 2787-2805, 2010.

[25] L. D. Xu, W. He and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on industrial informatics,* vol. 4, no. 10, pp. 2233-2243, 2014.

[26] A. Botta, W. D. Donato, V. Persico and A. Pescape, "On the Integration of Cloud Computing and Internet of Things," in *IEEE International Conference on Future internet of things and cloud (FiCloud)*, 2014.

[27] L. Pipino, Y. Lee and R. Wang, "Data quality assessment," *Communications of the ACM,* vol. 4, no. 45, pp. 211-218, 2012.

[28] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," *Computer Networks,* vol. 56, no. 18, p. 3825–3833, 2012.

[29] N. Tyagi and S. Simple, "Weighted page rank algorithm based on number of visits of links of web page," International Journal of Soft Computing and Engineering (IJSCE), pp. 2231-2307., 2012.

[30] M. Franceschet, "PageRank: Standing on the shoulders of giants," Communications of the ACM, vol. 54, no. 6, pp. 92-101, 2011.