Valid.IoT - a Framework for Sensor Data Quality Analysis and Interpolation

Daniel Kuemper, Thorben Iggena, Ralf Toenjes
Lab for RF-Technology and Mobile Communications
University of Applied Sciences Osnabrueck
Osnabrueck, Germany
d.kuemper@hs-osnabrueck.de
t.iggena@hs-osnabrueck.de
r.toenjes@hs-osnabrueck.de

Elke Pulvermueller
Institute of Computer Science
University of Osnabrueck
Osnabrueck, Germany
elke.pulvermueller@uni-osnabrueck.de

ABSTRACT
Heterogeneous sensor device networks with diverse maintainers and information collected via social media as well as crowdsourcing tend to be elements of uncertainty in IoT and Smart City networks. Often, there is no ground truth available that can be used to check the plausibility and concordance of the new information. This paper proposes the Valid.IoT Framework as an attachable IoT framework component that can be linked to generate QoI vectors and Interpolated sensory data with plausibility and quality estimations to a variety of platforms. The framework utilises extended infrastructure knowledge and infrastructure-aware interpolation algorithms to validate crowdsourced and device generated sensor information through sensor fusion.

CCS CONCEPTS
• Information systems → Sensor networks; Trust; • General and reference → Evaluation; Estimation; Verification;

KEYWORDS
Quality of Information, Crowd Sourcing, Sensor Fusion, Information Validation, Sensor Interpolation, IoT Framework, Spatiotemporal Data Analysis, Sensor Fusion

ACM Reference Format:

1 INTRODUCTION
The most valuable resources of information in smart city areas are its infrastructure and its citizens. However, the Internet of Things (IoT) and the Internet of People (IoP), i.e. social media or crowdsourced sensor information, are not seamlessly integrated information-wise. This information gap requires methods for integrating IoT and social network data that pay particular attention to the quality of the data. To find a common approach, this work proposes the Valid.IoT Framework, which utilises Quality of Information (QoI) metrics to determine the quality of heterogeneous information sources and generates QoI Vectors to annotate the sensor information. Furthermore, Valid.IoT refines the state of the art by utilising crowdsourced and device-generated sensor information combined with domain-independent correlation and interpolation models whilst incorporating knowledge of the city infrastructure to evaluate information plausibility and concordance.

The remainder of the paper is structured as follows: Section 2 presents the state of the art whereas Section 3 describes the motivation of an infrastructure knowledge- and propagation-driven quality evaluation framework. Section 4 gives a brief overview whereas experimental results are discussed in Section 5. The paper concludes in Section 6.

2 STATE OF THE ART
The assessment of information quality can basically be evaluated in five common dimensions: Completeness, Correctness, Concordance, Plausibility and Currency. [29] provides a table of different terms used to describe one of the dimensions of data quality. Furthermore, they provide a mapping between data quality dimensions and data quality assessment methods. [19] introduced Sieve, a framework to flexibly express quality assessment methods and fusion methods. Since Mobile Crowd Sensing (MCS) applications generate large amounts of sensed data, which is collected and preprocessed by devices with limited energy supply, challenges arise with respect to sensor management to ensure an energy-aware and quality-driven data acquisition process. [18] present a model for G-MCS and evaluate its energy savings for different application requirements and geographical sensor distribution scenarios. Semantic interoperability, as a prerequisite for platform cooperation, has been widely addressed in the literature. symbIoTe[31] goes a step further to propose novel aspects of organizational interoperability by introducing a concept of IoT platform federations and roaming IoT devices. These platform capabilities can be utilised to validate user and data patterns.

One of the major challenges in the assessment of a quality metrics of sensory IoT data is the lack of ground truth. This problem is well-known in the image processing domain when trying to rate the quality of a single picture without any reference. The task is often described as blind image quality assessment [17] or No-Reference Image Quality Assessment (NR-IQA) [20]. To get an objective quality metric NR-IQA analyses the sharpness of edges or the noise levels. While these approaches can be used to determine the quality of data they are unfit to make a statement about the
plausibility of the information, both in image processing and sensor networks.

The authors of [3] and [4] developed and evaluated a concept for the assessment of node trustworthiness in a network based on data plausibility checks. They propose that every node performs a plausibility check to identify malicious nodes sending faulty data. Similar to this work they use similar data sources in order to find witnesses for a given sensor reading. [27] presents a comparative analysis of trust (models and metrics) in diverse contexts, provides a comprehensive ontology to capture trust-related concepts and utilises Bayesian approaches to trust and illustrate its use in sensor networks.

The authors of [30] propose three different approaches to deal with a missing ground truth in social media: spatiotemporal, causality, and outcome evaluation. Their concept of using spatiotemporal evaluation to predict future behaviour of humans shows similarities to our Valid.IoT approach, which is additionally evaluating large amounts of historic event data. Prior work of the authors emphasised the importance of an appropriate distance model reflecting infrastructure, e.g., roads, and physics, i.e., traffic or air movements [28].

The authors of [1] show that spatial correlations need complex evaluation algorithms that also consider the temporal distance between events. Furthermore, the requirement of non-linear coherence/correlation determination is demonstrated (e.g., Mutual Information in contrast to common Pearson Correlations).

In recent work the authors of this paper pointed out [13][16] that in addition to temporal propagation, information correlation processing and information interpolation have to take the infrastructure into account. Various errors can slip into the information processing if algorithms utilise simple spatiotemporal distance metrics like the Euclidean distance instead of utilising infrastructure-based metrics that are entity bound. The utilisation of varying sensor stream QoI to dynamically change used data sources during the IoT framework runtime is shown by the CityPulse Project [21][14].

The proposed approach in Valid.IoT refines the state of the art by utilising sensor and domain independent correlation and interpolation models whilst incorporating knowledge of the city infrastructure to evaluate data stream plausibility and concordance.

3 MOTIVATION AND REQUIREMENTS

Continuous monitoring and evaluation of information quality are essential for an appropriate utilisation and early identification of failures in using IoT data/services to avoid misleading reasoning results. Furthermore, human interaction requires an intense evaluation of information quality, due to an easily challenged trustworthiness of automated smart decision systems. If a user is sitting in a car on a blocked road and an artificial intelligence claims that there is a free traffic flow, trust decreases rapidly.

Domain knowledge allows improving sensory information by use of related propagation and interpolation models. The idea of Valid.IoT is to provide generic models for data interpolation and extrapolation. Virtual sensors can offer additional information where no real-world sensor deployments are available (e.g. due to intermittent disruptions) or if a continuous collection of high-frequency real-world measurements is too resource intensive. Finally, the provision of reliable IoT services requires monitoring of quality metrics and the development of an active fault recovery mechanism.

The following sections describe the QoI-driven concepts behind Valid.IoT, as well as the idea of combining quality analysis and value interpolation. Furthermore, they describe an integration approach into existing IoT frameworks.

3.1 Quality Metrics

The quality and reliability of information, which is processed by IoT sensors, is one of the main criteria for users, applications, and machines to take decisions in the real-world. The Valid.IoT framework utilises a metadata annotation for IoT-related data attributes supporting structures related to quality metrics, based on QoI and Quality of Service (QoS). These metrics are used to rate the QoI from individual data sources based on their information plausibility and technical availability.

Therefore, the framework provides mechanisms for data attribute representation and comparison including a model-based quality analysis for IoT data sources. This contains a plausibility check to provide a QoI and QoS vector used by the ranking mechanisms that have already been evaluated in the CityPulse [15] project. To ensure a positive user experience, the context of user and application needs and the changes in the data/service and their environment have to be considered.

The definition and evaluation of relations between IoT data sources are essential to perform plausibility analysis and cross-validation of information provided by heterogeneous IoT sensors to form a multi-dimensional QoI and QoS vector. The attributes describe the technical performance of the IoT resources, the plausibility of the information, and accuracy of the measurements. Based on the developed propagation models and the probabilistic models, which are used in the Virtual Sensor (see Section 3.2), this component provides mechanisms and algorithms that are capable of modelling the correlations between data sources of the same type as well as relations between heterogeneous data sources. IoT devices are deployed in the real-world, which is dynamic and can relate to infrastructural changes over time. This means that their parameters change over time. Furthermore, the target applications and service requirements can change due to network/environment change. As Valid.IoT has to process information quality requests of various correlated data sources during runtime a scalable solution has to be provided. By using identified correlations between multiple IoT sensors and by monitoring the resource and network parameters and constructing scalable and high granular models, it is possible to adapt to individual changes without reconfiguration of the whole IoT framework. The following metrics are calculated in the Valid.IoT framework and can be used to annotate a dataset with a QoI vector.

The QoI Vector $\vec{Q}$ consists of five numeric values

$$\vec{Q} = (q_{cmp}, q_{tim}, q_{pla}, q_{art}, q_{con})$$

, whereby each individual value represents a normalised quality metric with each $q \in [0, 1]$. The following paragraphs describe the individual quality metrics and their detailed parameters.

Completeness $q_{cmp}$ defines if a data source message provides all information that was defined in its description. While registering a
new data source a part of its description are the fields that should be contained within delivered messages, e.g., temperature, humidity and wind speed for a weather sensor. The Completeness can then be calculated as

\[ q_{cmp} = 1 - \frac{M_{\text{miss}}}{M_{\text{exp}}} \]

with \( M_{\text{miss}} \) = sum of missing values and \( M_{\text{exp}} \) = sum of expected values of an incoming dataset. \( M_{\text{exp}} \) is though extracted from the model description of the IoT Metadata of the data source (see Section 4.1). Each data source message is rated for Completeness on its own without including older messages.

**Timeliness** \( q_{\text{tim}} \) rates if an observation was processed within a defined time frame before being delivered to the framework. In technical terms, it calculates the difference between the current time and a time stamp of the measured effect. If the difference is out of the defined range (if the observation is too old) the QoI metric Timeliness is lowered. In contrast to common QoS evaluation, the time-related quality metric depends on the availability of the (updated) information rather than the technical transmission.

The Timeliness quality is evaluated against the previously annotated source properties (in the IoT Metadata component, see Section 4.1). \( T_{\text{freq}} \) defines the maximum time interval in ms expected between two measured values. \( T_{\text{freq}} = 0 \) defines a solely event-based measurement transfer. The Frequency is calculated as

\[ T_{\text{freq}} = t(x) - t(x-1) \]

where \( t(x) \) is the time stamp of a received data source message. This time stamp is compared to the time stamp of the last message. \( T_{\text{freq}} \) can then be used to measure the Timeliness of a data source message by comparing it with the announced timing settings in the data source description. In comparison to the Frequency the Age metric measures how old a data source message is when arriving in the framework. It is calculated by the difference between the current time stamp and the time the message was created.

\[ T_{\text{age}} = t_{\text{now}} - t(x) \]

To normalize the Frequency and Age within an interval of 0 and 1 the Reward and Punishment algorithm, introduced in [7], is modified and integrated. This algorithm takes numerical values like Age or Frequency and compares them with some given upper and lower bounds annotated within the IoT Metadata description. If a measured value is within these bounds the reward increases otherwise it is punished. The reward is calculated as follows

\[ Rd(t) = \frac{a^{W-1}(t-1)}{W-1} + \frac{a^{W-1}(t-1) + a^{\text{current}}(t)}{W} \]

where \( W \) is the length of a sliding window over the last inputs and \( a^{W-1} \) denotes the number of measurements within the given interval. \( a^{\text{current}} \) \( \in \{0, 1\} \) is therefore the current reward or punishment decision which will be 1 for a measurement within the interval or 0 otherwise. The quality metric can then be calculated using

\[ q(t) = |q(t-1) - 2 \cdot Rd(t)| \]

with \( q(t) \) for the value of the quality metric at time \( t \) and \( q(t-1) \) for the past value.

By normalising both submetrics of Timeliness with the help of the Reward and Punishment algorithm they are combined together to form the numeric value \( q_{\text{tim}} \) for the Timeliness within the QoI vector \( \hat{Q} \).

**Plausibility** \( q_{\text{pl}} \) defines if a received data source information makes sense regarding the probabilistic knowledge about what it is measuring. Therefore, physical value ranges (e.g., indoor temperature or vehicle speed) and historic information are used to calculate the plausibility of a measurement.

**Artificiality** \( q_{\text{art}} \) determines the degree of used sensor fusion techniques and defines if this is a direct measurement of a singular sensor, an aggregated sensor value of multiple sources or an artificial spatiotemporal interpolated value. If the sensor information originates from an individual IoT hardware sensor, which is not aggregated or interpolated we assume \( q_{\text{art}} = 1 \). An unidentified information source, which aggregates information with unidentified algorithms will be annotated as \( q_{\text{art}} = 0 \). The metric can be individually adapted to the utilised openness of the connected IoT framework.

**Concordance** \( q_{\text{con}} \) is used to describe the agreement between information of the data source and the information of further independent data sources, which report correlating effects. The Concordance analysis takes into account a finite set of \( n \) sensors with their location \( x_i \) and their individual Concordance \( c(x_i, x_j) \) with \( i \in \mathbb{N} \) and \( c(x_0, x_1) \in [0, 1] \) as share of measurements that witness the sensor observation. The decision, which individual information agrees with each other (e.g., slow traffic event with \( \Delta v = -12 \text{km/h} \)) is stored in the IoT Relationship Model, see Section 4.1.

The concordance \( q_{\text{con}}(x_0) \) at the given sensor location \( x_0 \) is calculated by the concordance function

\[ q_{\text{con}}(x_0) = \sum_{i=1}^{n} \lambda_i(x_0) \cdot c(x_0, x_i) \]

\[ = \lambda_1(x_0) \cdot c(x_0, x_1) + \lambda_2(x_0) \cdot c(x_0, x_2) + \ldots + \lambda_n(x_0) \cdot c(x_0, x_n) \]

with \( \lambda \) as a weight-function

\[ \lambda_i(x_0) = \frac{1}{d(x_0, x_i)} \quad \text{if } d(x_0, x_i) \neq 0 \]

and \( d(x_a, x_b) \) as a propagation- and infrastructure-based distance function between sensor location \( x_a \) and \( x_b \) for sensor \( a \) and \( b \). It can be assumed that \( q_{\text{con}}(x_0, x_0) = 1 \), since it represents the same sensor and the distance is 0. To avoid the division by 0 the function is normalised as

\[ \lambda_i(x_0) = \frac{\lambda_i(x_0)}{\sum_{i=1}^{n} \lambda_i(x_0)} \]

To achieve an exponential neglection of samples, which have a high distance, the \( x_i^b \) power can be applied based on the derived propagation model.

\[ q_{\text{con}}(x_0) = \frac{\sum_{i=1}^{n} c(x_0, x_i)}{\sum_{i=1}^{n} d(x_0, x_i)^b} \]

**Interface**

The quality metrics of individual data sources can be requested in different levels, which are shown in Figure 1. Depending on the interest in the detail of the application and the assigned access rights the following levels can be requested:

- **lvl0** Atomic aggregated value of the quality vector \( \hat{Q} \): average, max, min, percentile.
The provision of a continuous spatial sensor coverage enables common IoT services to be used at any location in a specified area. Valid.IoT’s virtual sensor concept utilises domain- and infrastructure knowledge to verify IoT data and derive information for unobserved locations. Existing approaches try to determine and verify the correctness of IoT data and IoT services by using individual data sources or considering simple correlation models. The Virtual Sensor component extends these approaches by introducing novel propagation models, which also respect the physical properties of the real-world measurement, their dependencies to other physical quantities and how they influence each other. For example, traffic propagates on streets with a certain velocity, subway trains are only accessible at distinct stations and noise propagates with a topology-dependent attenuation. The integration of various effect propagation predictions enables a flexible adaptation of existing traffic-[9], noise-[2], pollution-[32] and social-propagation algorithms.

Heterogeneous IoT infrastructures are likely to lack sufficient sensor coverage for a sophisticated and reliable algorithmic exploitation for every scenario. Placing IoT sensors at every desired location in the entire IoT environment such as a city or an industrial complex is expensive and not feasible. To be able to get information from places without a dedicated IoT sensor, interpolations between information of nearby sensors can provide a better information coverage by using the developed propagation models. While a propagation model only describes the distribution of a measured phenomenon from the sensed location and a correlation model describes relations between data sources, an interpolation model can estimate new physical properties when, due to intermittent conditions, a resource becomes unavailable. Obviously, this will not be as accurate as the deployed resources. Therefore, the main concept of Valid.IoT is the description of the QoI of every data source since no resource is accepted as a static ground truth. Every existing information source will have an impact on the quality of the virtual data sources’ data. This will lead to a sensor network composed of virtual sensors as an overlay of the existing one. By utilising infrastructure[16] knowledge and previously derived phenomena dependency models, we can provide more sophisticated interpolation methods than common approaches.

Furthermore, the inference of new virtual data sources and the calculation of the virtual source QoI vector provide guidance for the deployment of new IoT devices since locations with low information quality can easily be identified. By evaluating the probabilistic plausibility and concordance of interpolated sensor information, it is possible to determine critical locations where real sensors are still needed. This results in a cost-effective sensor roll-out strategy.

The interpolation models are based on logistics and linear regression models and also mutual entropy models to infer virtual data sources by using the existing information of related data sources. Piecewise constant interpolation, as one of the simplest interpolation methods, is an inappropriate interpolation method with jump discontinuity. Advanced interpolation methods i.e., Kriging and Inverse Distance Weighting (IDW) are extended with context and infrastructure information and propagation parameters to calculate the Infrastructure-based Inverse Distance Weighting (I-IDW) [16]. They consider the fading, the direction and other properties of the existing and new physical phenomenon to choose the best interpolation method regarding precision and computational costs. A probabilistic relationship model is defined and utilised to infer information from sources to create the virtual devices.

### 3.3 Enablement of Monitoring and Fault Recovery

Changes in IoT environments result in non-optimal service operations or operation failures and downtimes. The resources need to be published with sufficient information about their QoS and QoI metrics and also historical QoS/Qol information. The IoT resource/data failures can be caused by many different reasons: hardware, network, device, software or human/environmental related issues. It is important to identify possible faulty information sources before the actual unqualified information is propagated in the system. By continuously monitoring their QoI/QoS properties and providing alternate information sources, a preventive fault recovery mechanism can be provided. A sensor fault recovery strategy has to decide if sufficient information is available and has to provide the needed quality. Furthermore, it indicates if new virtual resources have to be set-up to compensate for a (temporarily) unavailable resource with a sufficient quality.

The Runtime Monitoring for IoT-based Services enables a scalable monitoring solution with integrated fault recovery mechanisms to improve the quality of IoT data provided to higher-level applications and to achieve improved service continuity [15]. Changes in the quality are captured in the QoI/QoS values of the resources. The ecosystem intelligence of an external IoT framework is provided by accessing and analysing the QoS and QoI information, discovering alternate resources, and using a monitoring and control plane for another network/system. Supervised machine learning classification technologies (e.g., Linear Discriminant Analysis, Bayes and Support Vector Machines) can be utilised to enhanced propagation and interpolation models. This knowledge can be used to achieve an optimised fault recovery.

### 4 FRAMEWORK COMPONENTS

This Section describes the basic structure of the Valid.IoT Framework, which is shown in Figure 2. It is basically partitioned into...
three packages. The Sensor and Relationship Modelling describes individual information sources, their technical implementations and relationships between each other. The Validation and Interpolation uses the previously defined models to analyse data streams in order to enable Information Quality Analysis and Virtual Sensor provision via Interpolation. The IoT Framework Interlink enables connectivity to existing IoT frameworks and enables Monitoring and Testing functionalities. The following paragraphs give a detailed description of the framework packages, components and models.

4.1 Sensor and Relationship Modeling

The IoT Relationship Model (IRM) describes the relationship between individual sensor entities in the real-world and their mutual impact. This relationship can depict that e.g., sensors, which are in the same room, are connected by any physical relationship (e.g., different temperature sensors, placed in a river) or are connected via an infrastructure like traffic on a road network. It utilises the spatiotemporal propagation model of a data source to describe the interconnection and propagation of sensor values based on their relation. However, the model does not necessarily describe inevitable effects between data streams. If, for example, a traffic sensor $A$ reports an average car count of 0 vehicles per minute, a data source $B$ that reports traffic jams out of different sensor data can be used to verify, respectively support the concordance of the information of sensor $A$. At the same time, 0 vehicles per minute does not necessarily deduce a traffic jam. There could also be no traffic during the night.

The Infrastructure Model stores the physical models that can be used to determine the relations between the IoT data sources and actuators. It stores e.g., maps and building plans, which can be used to determine propagation directions. As an example, the OpenStreetMap database is used to get infrastructure information for propagation on roads, pedestrians and trains.

Figure 3: Infrastructure Model Description of the Utilised OpenStreetMap Database[12]

Figure 3 shows an example of the detailed information, which is used to restrict the propagation of vehicle-related propagation patterns. It shows the definition of road type, e.g., $a$: $h$=secondary is the definition of a road category similar to a main road (see. [6]) and the definition of junction points, which connect the road segment,
and areas like cycleways. The OpenStreetMap infrastructure model is stored as a directed graph, which is tagged with a defined set of attributes for every edge and node of the graph, which e.g., define maximum speeds, one-way usage and the road surface.

The **Interpolation Algorithm** defines possible algorithms to determine spatiotemporal interpolation of sensor values based on individual Propagation Model and Infrastructure Model. Simple areal propagation scenarios are solved using interpolation algorithms like Kriging and IDW. However, the novel approach of Valid.IoT is to regard restricting infrastructures and use propagation algorithms like the previously published (I-IDW) [16]. Furthermore, 3-dimensional gas-, dust- and noise propagations can be used on the same infrastructure model and use the digital terrain model or digital surface model (including buildings and other objects).

The **IoT Data Source (IDS)** describes the individual, technical data sources of sensors and data portals. It describes push/pull mechanisms, update intervals, and technical access patterns. It consists of Live Data (e.g., Sensors and Gateways) and Historic Data (e.g., databases and data portals) connections, which can refer to the same sensor entity. Furthermore, it describes data communication and processing including network dependant QoS parameters, such as throughput, latency, jitter, errors and availability of the data endpoint.

The **IoT Metadata** describes the measured or interacted real-world interaction of the measured scalar. It links technical descriptions and values to higher level information.

The **Spatiotemporal Propagation Model** models the individual propagation pattern of measured effects to enable a sophisticated sensor fusion. It provides medium-based propagation speeds and a link to the utilised Infrastructure Model [10] [8] [26].

The **Data Description** describes the individual mapping of real-world effects to the given data types and documents.

The **Information Description** describes the higher level information. It uses upper-level ontologies to explain the measured effect, e.g., water temperature, traffic speed or noise level.

### 4.2 Validation and Interpolation

The **Spatiotemporal Data Analysis** bundles the quality estimation and value interpolation and allows them to use the same models and interfaces. It analyses the **IoT Relationship Model** and uses the **IoT Relationship Compiler** to generate a trained model.

The **Information Quality Analysis** is a component of the Spatiotemporal Data Analysis and utilises the trained relationship models to estimate the quality of new datasets and generate the QoI vectors based on the defined metrics.

The **Propagation and Interpolation Calculation** is a component of the Spatiotemporal Data Analysis. It utilises the Infrastructure Model, Spatiotemporal Propagation Model and Interpolation Models to estimate effect propagations of a given sensor. As an example, the interface `getI(IRM, SensorEffect, Time, Location, Direction)` can be used to get the propagation of traffic congestion if a traffic sensor is defined in the IRM. Furthermore, it can be used for an aerial pollution propagation if a wind sensor and a gas emitter are defined in the IRM. Location and Directions are optional parameters which can request an interpolated value for a single location. If these are omitted, a full propagation map of the requested effect is created.

The **Virtual Sensor Provision** is a component of the Spatiotemporal Data Analysis and utilises the compiled and trained IRMs to continuously estimate interpolated values based on the known propagation models in the given infrastructure. It creates a service endpoint in the Valid.IoT Core, which constantly provides an estimated value or probability distribution and a QoI vector, which annotates the concordance.

The **IoT Relationship Model Compiler** enables a joined access for Plausibility and Interpolation models for trained data scenarios. It enables the provision of these models for the Spatiotemporal Data Analysis as well as updates of datasets to retrain time series- and neuronal network-based approaches.

The **Time Series Analysis** enables a time series decomposition-based analysis and regression-based interpolation.

The **Machine Learning** enables a machine based learning approach. It generates compiled and pre-trained models, which can be used to predict information plausibility and a probabilistic value prediction.

### 4.3 IoT Framework Interlink

The **Runtime Monitoring for IoT-based Services** focuses on mechanisms to collect and process metadata about the resources in the form of dynamic information about their QoI and QoS attributes to respond to concept drifts and data drifts. It allows to include the monitoring and management information and provide early identification of faulty IoT devices/services to provide appropriate adjustments for the evaluation algorithms. It provides the **Valid.IoT Core** with datasets from the **External IoT Framework**. The **Test Oracle for IoT-based Services** acts as a component for model-based testing approaches [23], which will be presented in future publications.

### 5 EXPERIMENTAL RESULTS

For a basic evaluation of the Framework functionality the datasets, described in Table 1, have been used. Open Data Aarhus (ODAA) is an open data portal for the Aarhus municipality and allows access to various datasets, e.g., traffic and cultural event data. The traffic data consists of 449 deployed traffic sensors in the city, which report information about the vehicle count and average speed on the defined sensor location. Only a subset of driving cars is measured since these traffic sensors only measure activated WiFi/Bluetooth communication devices. During the CityPulse project [22] this data has been collected for more than two years and is available as a semantically annotated dataset.

The second data source from TomTom allows access to a set of developer APIs that delivers traffic information. The Traffic Flow API provides information about the current average vehicle speed and the typical vehicle speed of a requested area. The Traffic Incidents API delivers information about registered events, categorised into 4 Levels: 1 Slow Traffic, 2 Queuing Traffic, 3 Stationary Traffic, 4 Closed Road. This traffic information is generated from movement of mobile devices (navigation apps, in-dash navigation, phones [5] [25]).
Table 1: Examined Traffic Datasets

<table>
<thead>
<tr>
<th>Source</th>
<th>Dataset</th>
<th>Spatial Resolution</th>
<th>Provision</th>
<th>Frequency</th>
<th>Information</th>
<th>Sensor</th>
<th>Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ODAA</td>
<td>Traffic</td>
<td>Road Segment</td>
<td>Periodic</td>
<td>1/5min</td>
<td>Flow</td>
<td>Speed (km/h)</td>
<td>Agg.(Mean)</td>
</tr>
<tr>
<td>Cultural-Events</td>
<td>Event Location</td>
<td>Periodic</td>
<td>1set/day</td>
<td>Capacity</td>
<td>Max. Visitors</td>
<td>Agg.(Sum)</td>
<td></td>
</tr>
<tr>
<td>TomTom</td>
<td>Traffic Flow</td>
<td>Road Segment</td>
<td>Periodic</td>
<td>1/30min</td>
<td>Current Speed</td>
<td>km/h</td>
<td>Agg.(Mean)</td>
</tr>
<tr>
<td>Traffic Incident</td>
<td>Coordinate/Direction</td>
<td>Event-Based</td>
<td>-</td>
<td>-</td>
<td>Set of Incidents</td>
<td>Severity (4 levels)</td>
<td>Reasoned</td>
</tr>
</tbody>
</table>

Both data sources are measuring just a subset of the cars that are driving on the road of interest. Therefore, they are not able to deliver distinct and precise information of all vehicles in the city. Figure 4 shows the fragmentary measurement of the different data sources. Each data source only covers a subset (red or blue) of the existing vehicles, although the detections may overlap (red-blue gradient). A large subset of vehicles may not be tracked (white). Therefore, a full model of all vehicles and their exact movement through the city is not available with common technology. The following section shows that a cross-evaluation of both datasets, which are generated out of two independent sensor sets, is a useful measure to determine their concordance.

5.1 Time Series Analysis Validation

Since continuous smart city data is often based on seasonal patterns and appears as noisy time-based datasets the ability to separate trend, seasonality and irregular components allows an interpretation of the current situation. For example, when there are no cars measured on a small street at night, it does not typically mean that there is a traffic jam. Therefore this evaluation applies a decomposition of the time series into additive seasonal, and irregular components using moving averages.

5.2 Event-Based Results

This section evaluates the concordance between ODAA and TomTom data sources by analysing 8607 incidents, that have been captured by the authors with the TomTom API for Aarhus, against the ODAA sensor data set (six weeks time series data for up to nearest 10 sensors). The overall results show that for more than 75% of the severity level 4-closed the raw data of a nearby ODAA sensor measured a vehicle count of 0 cars during the event period. Figure 5 shows the cumulative distribution function (CDF), of the irregular component of the ODAA during the TomTom incidents. It shows that for reported events most vehicle speed readings are slower than described by the seasonal component $S_t$, which means that the irregular component $I_t < 0$. The change of the vehicle count does not clearly reflect the event. The vehicle count can drop because of a traffic jam or a traffic jam can be the result of the road being overloaded. These measurements show a high concordance value of measured slow, queueing, stationary traffic incidents compared with the irregular speed component of the traffic sensors. However, the concordance of the closed events can only partially be acknowledged. With around 30% having a $q_{con} = 0$, due to non-concurring observations, the overall data source concordance is $q_{con} = 0.69$.

5.3 Continuous Measurement Result

After the discussion of event-based sensor readings in the last section, this section deals with measurements on a continuous scale. Therefore, TomTom Traffic Flow data has been evaluated against
ODAA traffic flow measurements. For every TomTom measurement that showed a severe slowdown of more than 15km/h compared to the free flow speed an evaluation has been triggered: Six weeks time series data for up to 10 sensors within a maximum distance of 200 meters have been analysed and their minimum values of the irregular component have been compared to the TomTom measurement. Figure 6 shows the heatmap distribution of 28306 pairwise comparisons of the two data sources. The colour in the heatmap shows the density of measurement points. The figure depicts that there is no clear linear correlation between the two data sources. 95.5% of ODAA comparisons also have an irregular component value $I_r \leq 0km/h$, which is indicating a decrease in speed. These measurements confirm the measurement of the evaluated sensors and determine the concordance $q_{con} = 0.96$ of the TomTom Flow data, although there is no direct correlation between measured values.

5.4 Machine Learning

An alternative quality estimation and prediction scenario was implemented in the machine learning environment Keras[11]. The experiment used historical sensory information and infrastructure meta data (e.g., direction, distance, time parameters) as input for the individual neural networks. The next section describes an approach to find a concordance estimation of a measured value. The subsequent section describes an approach to use the same data to estimate a single value at a dynamic location.

5.5 Measurement Validation

In this experiment, a neural network was created to validate new sensor values of the previously mentioned traffic sensors in Aarhus. Therefore, the network is fitted with spatiotemporally correlated data from nearby sensors. Figure 7 shows the model of the neural network. It was designed to predict an output vector of 151 speed classes (mapping to $0 – 150 \frac{km}{h}$ average vehicle speed). By using the Softmax activation function a Probability Density Function (PDF) is created to show 151 individual probability classes with a cumulated probability of $\sum_{x=0}^{150} p(x)dx = 1$. The previous Dropout layer was designed to prevent the model from overfitting. The first layer after the normalised input layer was activated with a Rectified Linear Unit. The input vector (same weights) consisted of decomposed temporal attributes (e.g., day of week, day of year, hour, minute) to allow the recognition of repeating temporal patterns, as well as the spatial attributes (e.g.: latitude, longitude and road direction).

Training a model with 7370929 data sets of 25 traffic sensors (3 years of data) takes 11 minutes with GPU usage$^2$. Training parameters have been:

- Optimizer=RMSprop
- Loss=Binary Cross-Entropy
- Metrics=Accuracy
- Learning data: 80%
- Validation data: 20%

Predictions of the probability patterns (see Figure 8) took less than 0.1ms per pattern during runtime of the framework.

Figure 8 shows some exemplary probability density functions (PDF) for traffic sensor evaluation and prediction. The x-axis depicts different speeds $\nu = [0, 150] \frac{km}{h}$. The y-axis shows the probability of each individual integer with $\sum_{x=0}^{150} p(x)dx = 1$. Furthermore, the 90th and 50th percentile are plotted (see Legend Figure 8). The results after 10 Backpropagation epochs: $loss = 0.0274$, $accuracy = 0.9941$, $validationloss = 0.0258$, $validationaccuracy = 0.9945$. This acknowledges a non-overfitted model, which is able to rapidly predict the plausibility of Aarhus Traffic. A manual evaluation with measured values that did not fit in the prediction often showed malfunctions in sensor nodes.

5.6 Value Prediction

In difference to the probability vector of the previous section this experiment aimed at estimating a single value for a missing sensor or missing sensor value. Therefore, similar to Figure 7 a new neural network was created using the same input-vector but just a 1-dimensional output vector, predicting the average speed value with a Sigmoid activation layer. This time we used 1 Year of data

$^2$ Intel i7-5820K, with Nvidia GTX1080Ti
for all 449 sensors also with 80% learning data and 20% validation data. Figure 9a shows the CDF of the absolute measured speeds in the array of the target (label) data that was used to train/evaluatethe neural network. Figure 9b shows the CDF of absolute speed differences of the predicted values by comparison to the measured values (\( |\text{measured} - \text{predicted}| \)). It shows the comparison of 3 different approaches (see legend), which are explained in the following:

**accuracy rmsprop**: Neural Network, 16 Inputs, 1 Output, Optimizer=RMSProp, Loss=Binary Cross-Entropy, Metrics=Accuracy

**last_value**: No neural network. Just the difference to the last measured value of the sensor as a comparison to the most common approach.

**accuracy rmsprop_noinf**: Neural network configuration of accuracy rmsprop without the spatial input vectors (only 11 input vectors)

Overall, Figure 9 shows that the neural network with spatial knowledge (accuracy rmsprop) provides the estimations with the lowest error. Figure 10 shows the correlation between the error and the measured value. 50% of the error is less than 2.30 km/h and 73% less than 5 km/h, which is a very low absolute error for a prediction that would not affect decisions. 10% of the estimations have more than 11.96 km/h deviation from the measured value, which can lead

that there are existing congestions or a that congested road is free.

1% of the estimations have more than 33.61 km/h deviation from the measured value, which gives a very wrong expression of the current situation and could lead to wrong decisions.

Although the predicted values have some massive miss-predictions, the experiment showed that a neural network with spatiotemporal knowledge gives the most sophisticated estimations.
CONCLUSION

This paper discusses measures for evaluating and ensuring the quality of IoT’s sensory data streams. It provides a model-based sensor relationship approach and is integrating new evaluation schemes. This approach supports appropriate spatiotemporal distance measures to achieve an efficient monitoring of relevant correlating data. The spatial infrastructure exhibits a high impact on the interdependency of sensors. While the temperature will be similar in the neighbourhood, noise propagation will depend on shielding buildings and traffic flows depend on road networks, on-going construction work, traffic density etc. Hence, spatial reasoning requires appropriate distance measures that are based on the adapted propagation model.

In conclusion, the suggested framework provides methods to cope plausibility analysis for heterogeneous data sources in IoT applications, and in addition, considers that the sensors become unreliable over time by approaching a continuous time series and machine learning evaluation. Furthermore, it enables the operation of a virtual sensor layer, which provides interpolated information with attached quality vectors. In the future, we aim at optimising the framework by integrating the infrastructure assembly through projection onto convolutional neural network layers.

ACKNOWLEDGEMENT

This work has been supported by the European Union Horizon 2020 project IoTcrawler under grant agreement number 779852.

REFERENCES


All online resources have been finally accessed on 2018-02-02.