Future Generation Computer Systems 149 (2023) 701–716

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

A big data exploration approach to exploit in-vehicle data for smart road maintenance

Devis Bianchini *, Valeria De Antonellis, Massimiliano Garda

University of Brescia, Department of Information Engineering, Via Branze, 38, Brescia, 25123, Italy

ARTICLE INFO

Article history:
Received 6 December 2022
Received in revised form 1 June 2023
Accepted 4 August 2023
Available online 9 August 2023

Keywords:
Big data exploration
Data streams summarisation
Multi-dimensional model
Smart and resilient mobility

ABSTRACT

In modern Smart Cities, pervasive collection of sensor-based and IoT data streams is a challenging opportunity for improving mobility resilience. Among the potential applications, sensor-based data streams provide valuable information about the quality of the area-wide road surface. Modern vehicle black boxes are also able to estimate the type of anomaly (e.g., bump, hole, rough ground, depression), based on real-time analysis of acceleration data streams. Road maintainers may use all this information to improve monitoring and maintenance activities. However, the volume of data streams, the variety of road network and different degrees of seriousness of detected anomalies call for methods to support maintainers in the exploration of available data. To this aim, in this paper, we propose a methodological approach, based on big data exploration techniques. The approach is grounded on: (i) a multi-dimensional model, apt to organise data streams according to different dimensions and enable data exploration; (ii) data summarisation techniques, based on an incremental clustering algorithm, to simplify the overall view over massive data streams and to cope with their dynamic nature; (iii) a measure of relevance, to focus the attention on road portions that present critical conditions. The innovative contributions regard the formalisation of the exploration methodology, the definition of exploration scenarios, based on road maintainers' goals and the measure of relevance, and an extensive experimentation on a real world case study, addressed in a research project on smart and resilient mobility. Experimental results show how relevance evaluation is able to efficiently attract the road maintainers' attention on road portions that present the most critical conditions and the proposed incremental clustering algorithm outperforms existing algorithms in the literature.

1. Introduction

In the latest years, the increasing availability of big data has become a key factor in shifting towards a data-centric vision of modern Smart Cities [1]. In particular, the concept of Smart Mobility and its impact on the logistics of transporting goods and people are experiencing radical changes, capitalising on big data generated from sensor networks and IoT devices [2]. Issues that can arise may be promptly noticed and tackled, increasing the quality of delivered services [3]. Among the potential applications, sensor data in vehicles may generate a continuous flow of data streams to provide valuable information about the quality of the area-wide road network. Modern vehicle black boxes are also able to estimate the types of anomalies detected on the road surface (e.g., bumps, holes, depressions, rough ground) by analysing in (near) real-time the acceleration traces. This information may be used by road maintainers to improve monitoring and maintenance activities, for enhancing mobility resilience. However, the volume of data streams, the variety of road network and different degrees of seriousness of detected anomalies call for methods to support road maintainers in the exploration of available data [4]. Therefore, road maintainers should be equipped with valuable analytical and exploratory tools, to gain insights from the data and ensure a safer and more efficient infrastructure.

To support road maintainers in monitoring road surface conditions and thus properly planning maintenance activities, we propose in this paper a methodological approach to foster big data exploration. The approach is grounded on three components: (i) a *multi-dimensional model*, apt to organise data collected on the road network according to different facets (based on features such as the type of road, area/district, mileage extent) and to enable data exploration; (ii) a *data summarisation algorithm*, that provides a synthetic representation of data streams gathered by vehicles, to simplify the overall view over massive data streams and to cope with their dynamic nature; (iii) a *measure of relevance*, aimed at focusing the road maintainers' attention on portions of the road network that present the most critical conditions.

The preliminary formalisation of the approach has been provided in [5], where the definitions of the three aforementioned

* Corresponding author.
E-mail address: devis.bianchini@unibs.it (D. Bianchini).

https://doi.org/10.1016/j.future.2023.08.004
0167-739X/

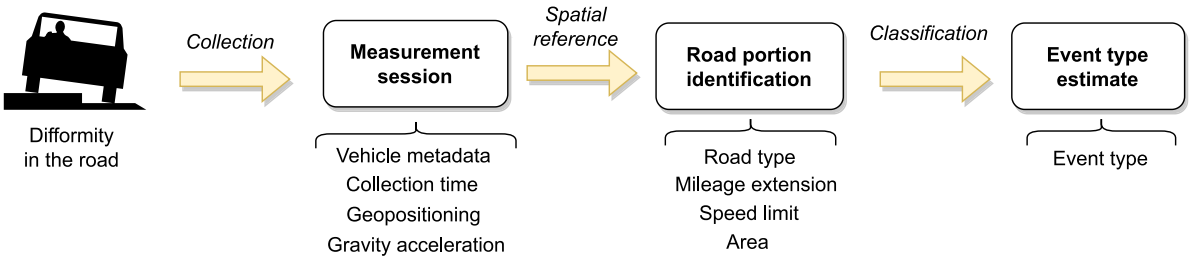

Fig. 1. Steps of the data collection procedure for monitoring the conditions of road portions.

components have been provided and declined in the context of sustainable and resilient mobility. The original contributions to the state of the art provided here concern: (i) a substantial revision of the measure of relevance, in order to be as more independent as possible from the maintainers' expertise in the preparation of the exploration parameters (this aspect emerged as a drawback in [5] and is worsened due to the variety of road network); (ii) a new formulation of the *exploration scenarios*, to be compliant with the new measure of relevance; (iii) the implementation of the exploration approach in a prototype tool, supported by a GUI, to let road maintainers execute exploration scenarios; (iv) an extensive experimentation on a real world case study, addressed in a research project on smart and resilient mobility. Indeed, exploration scenarios are conceived to move towards a human-in-the-loop data analysis vision [6], in which humans are actively engaged in the data exploration experience, with the goal of extracting useful insights from data.

The paper is organised as follows: in Section 2 the application context and the research challenges in mobility data exploration are introduced; Sections 3–5 present the data exploration methodology; Section 6 and Section 7 describe implementation issues and the experimental evaluation, respectively; in Section 8 a discussion about related work is provided; finally, Section 9 closes the paper, sketching future research directions.

2. Motivations

2.1. Application context

The proposed big data exploration approach is being applied in the scope of the MoSoRe research project,[1] whose aim is to investigate different perspectives on the resilience of mobility systems and infrastructures in the city of Brescia (Italy). In the MoSoRe project, a fleet of representative vehicles has been equipped with black boxes, to collect different kinds of data, ranging from acceleration traces in the form of data streams to an estimate of the type of anomaly that is recognised on the road surface.

Fig. 1 illustrates the steps behind the data collection procedure. The black box of a vehicle starts a *measurement session* when a sharp variation of the accelerometric measures is detected. During the measurement session, data collection is performed including the GPS coordinates and data streams from measures of accelerometers mounted on-board. Each data sample (measure) in the stream contains: (i) the timestamp; (ii) the GPS coordinates (latitude, longitude) of the vehicle; (iii) the accelerometers values over the three axes X, Y and Z. Furthermore, vehicle metadata such as its type (e.g., either commercial or private) and information related to the vehicle when the measurement session has been performed (e.g., average speed, direction) are associated with the data stream.

[1] Italian acronym for "Mobilità Sostenibile e Resiliente", funded by Lombardy Region (Italy), POR FESR 2014–2020.

The next step concerns the *identification of road portions* to be monitored, deemed as road segments sharing similar characteristics, such as the type (e.g., urban or suburban), containment area of the city (e.g., the district or other administrative divisions), mileage extension and speed limit. The GPS information enables the assignment of data streams to the *road portion* wherein the measurement session occurred.

Finally, the black box calculates a real-time *anomaly estimate* based on the accelerometric measures, thus associating the data stream with an *event type*. Currently, in the MoSoRe project, only four event types are recognised by the black boxes: hole, bump, rough ground, depression. In this respect, future efforts will be devoted to setup a classification system to enable the identification of a broader set of event types associated with detected anomalies. For example, in the scope of the project, on-board cameras are planned to be mounted on a group of representative vehicles, to be employed by road maintainers to collect and inspect video captures, thus recognising further event types, complementing the anomalies identified by black boxes.

2.2. Research challenges in mobility data exploration

Safety and comfort are the main features that users generally demand from a road infrastructure, looking forward to their continuous improvement over time by incorporating new technologies for the benefit of passengers, drivers and vehicles. Recently, intelligent transportation systems have been developed to improve traffic flow, road quality, safety levels, and availability of information to users. In particular, we focus on road maintenance solutions to help maintainers monitoring road surface conditions and properly plan maintenance activities. This paper addresses data exploration challenges mainly related to the problem of providing techniques to draw the attention of maintainers on relevant data collected from vehicles on the roads. Maintainers must be supported in the exploration of the large quantity of collected data and their attention must be attracted only on relevant data, corresponding to "anomalous" working conditions. The challenging issue is how to identify relevant data, considering that data relevance cannot be known before data is collected and stored. In addition, as they explore the collected data streams, maintainers may pursue different goals to monitor road status. Depending on the goal, road maintainers should be suggested to explore only the relevant portions of the road to enable management of critical situations. The considerations introduced above are translated into the following three challenges. For each challenge, techniques proposed to address it are specified as well, and described in details in Sections 3–5.

Handling multiple interleaved perspectives for data exploration. Given the complexity of the city road network, the enormous amount of data streams collected by vehicles should be interpreted by considering multiple aspects, such as the type of the road (e.g., urban, suburban), the administrative division (e.g., district, area), the mileage extension and the speed limit.

This intricate combination of data streams and dimensions must be properly made explicit during data exploration to take the right decision. Data modelling according to "facets" or "dimensions", either flat or hierarchically organised, has been recognised as a factor for easing data exploration, since it offers the opportunity of performing flexible aggregations of data, as already demonstrated in faceted search [7]. Therefore, a *multi-dimensional data model* designed for data streams is adopted to organise data portions in terms of dimensions (e.g., type of the road, administrative division, mileage extension, speed limit), at different granularity levels, to guide data exploration.

Reducing data massiveness of data streams to explore. The volume of data streams in the considered application domain might hamper the effectiveness of exploration and calls for efficient summarisation techniques, to provide a synthetic representation of the temporal evolution of road surface conditions. Moreover, while monitoring the conditions of a road portion through the observation of a physical phenomenon of interest, single measures might be affected by noise and false outliers (e.g., due to sensing malfunctions). In order to manage and explore large quantity of data, incremental clustering algorithms have emerged as promising solutions when treating data arriving at high rates, ensuring the possibility of retaining only lossless summaries of data, which are referred to as *syntheses* (or, with other analogous terms, *micro-clusters* or *Cluster Features* [8]). In our approach, we propose an algorithm articulated over two sub-tasks:

(1) *Generation/update of collected data summaries* – The clustering algorithm is applied to summarise data collected in a measurement session, in an incremental way; to capture the temporal evolution of summarised data streams, each run of the clustering algorithm generates/updates a set of syntheses, which is referred to as *snapshot*, and represents a status of the monitored road portion in a given time window;
(2) *Multi-dimensional organisation of snapshots* – Snapshots produced over time by the execution of the incremental clustering algorithm form a *sequence of snapshots*; snapshots in a sequence are organised according to the different analysis perspectives of the aforementioned multi-dimensional model and, therefore, are attributable to a specific road portion; road maintainers may perform an iterative exploration over a (sub-)sequence of snapshots, inspecting a specific portion of the summarised stream, thus easing data exploration.

Evaluating the relevance of data for effective exploration. The attention of road maintainers must be attracted only on relevant sequences of snapshots, inherently identifying road portions worth to be inspected in order to let road maintainers manage critical situations. This allows to alleviate road maintainers to rely too much on their expertise, focusing at the same time the exploration of road portions at the proper level of granularity (according to the organisation of snapshots imposed by multi-dimensional model) meeting the goals of road maintainers. Moreover, relevant road portions may be explored by road maintainers pursuing different goals, for instance: (i) given a sequence of snapshots representing an anomalous event, monitoring the evolution of such event over time; (ii) comparing sequences of snapshots related to anomalous events of the same type, thus establishing a seriousness prioritisation among them. To this purpose, a data relevance evaluation metric, combined with clustering-based summarisation, is required as independent from road maintainers' expertise and adaptable to different data exploration goals.

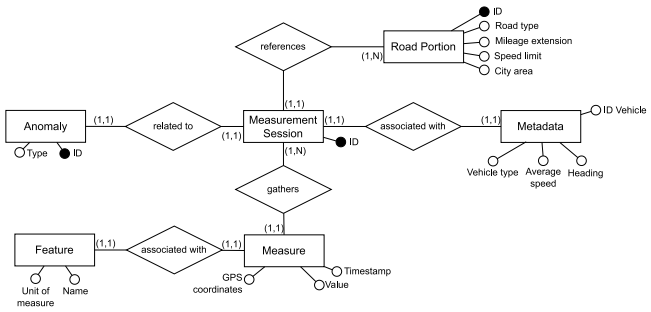

Fig. 2. E-R conceptual model of mobility data.

3. Multi-dimensional organisation of data

To support multi-perspective exploration of data streams, a Multi-Dimensional Model (MDM) has been conceived, grounded on dimensions and combination of dimension instances into exploration facets. In the following, we introduce the main pillars of the MDM for organising and exploring vehicles data streams, descending from the conceptual model in Fig. 2.

Features and Measures. Within a measurement session, physical quantities recorded by black boxes are referred to as *features*. A *feature* $f_i \in \mathcal{F}$ is a measurable quantity described by a name n_{f_i} and a unit of measure u_{f_i} (if any), where $\mathcal{F}$ denotes the overall set of features. The gravity acceleration over X, Y and Z axes, with m/s^2 as unit of measure, are examples of features.

A *measure* $x_i(t, GPS_{coords})$ is a scalar value for the feature $f_i \in \mathcal{F}$, expressed in terms of: (i) the unit of measure u_{f_i}; (ii) the GPS coordinates (latitude and longitude) GPS_{coords}, providing the position of the vehicle when the measure has been taken; (iii) the timestamp t.

Road portions. Portions of the road network are described with the properties introduced in Section 2.1, namely the type of the road (e.g., urban, suburban), the administrative division (e.g., district, area), the mileage extension and the speed limit. These properties are used to model the concepts of *dimension* and *exploration facet* as defined in the following.

Definition 1 (*Dimension*)**.** A dimension d_i is an entity representing a property of a road portion defined on a categorical domain $Dom(d_i)$. We denote with $\mathcal{D} = \{d_1, \ldots, d_p\}$ the finite set of dimensions. An instance $v^j_{d_i}$ of $d_i \in \mathcal{D}$ is a categorical value belonging to $Dom(d_i)$, $\forall i = 1, \ldots, p$ and $\forall j = 1, \ldots, |Dom(d_i)|$. Dimensions are organised into hierarchies $\{h_1, \ldots, h_m\}$, where each h_k is described with: (i) a subset of dimensions $Dim(h_k) \subseteq \mathcal{D}$, $\forall k = 1, \ldots, m$; (ii) a total order $\succeq_{h_k}$ on the elements in $Dim(h_k)$. □

Definition 2 (*Exploration Facet*)**.** An exploration facet ϕ_i (or, in short, facet) is a combination of dimension instances. Such a group of dimension instances is apt to identify road portions sharing the same characteristics. Let $\Phi = \{\phi_1, \ldots, \phi_k\}$ be the set of available facets. The cardinality of the set Φ ($|\Phi|$) spans over all the possible combinations of dimension instances, that is, $|\Phi| \leq 2^N - 1$, where $N = \sum_{i=1\ldots p} |Dom(d_i)|$, excluding the empty set combination and, generally, non combinable dimension instances.

Example. In the MoSoRe project, the dimensions considered for grouping road portions are *Type*, *Area*, *SpeedLimit*, *District*, *MileageExtension*. For example, *Urban* $\in$ *Dom*(*Type*) and *District1* $\in$ *Dom*(*District*) are sample dimension instances. A sample facet

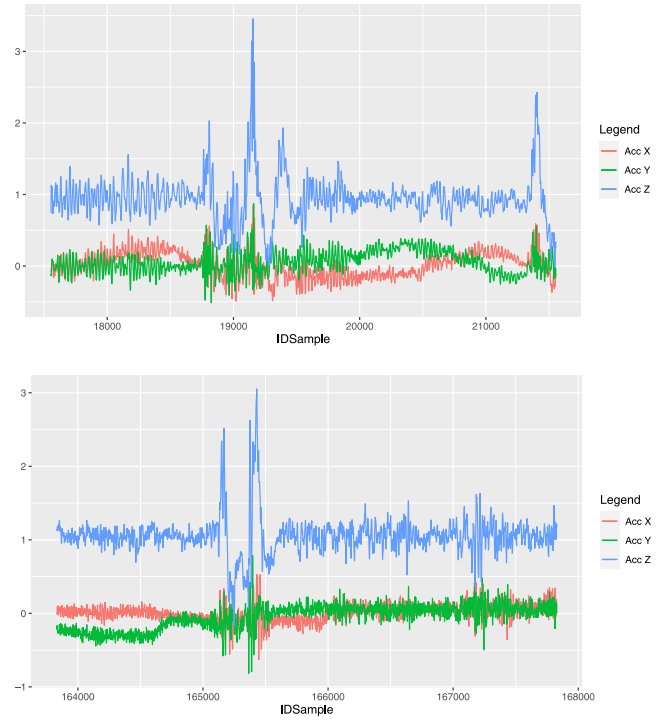

Fig. 3. Sample streams of measures (X-Y-Z accelerations) recorded by a black box in the case of a depression (top) and hole (bottom) for a specific road portion.

$\hat{\phi}$ may have as instances for the former dimensions *Type* = *Urban*, *District* = *District1* (*Area* = *South*), *MileageExtension* = $\leq$ 5 km and *SpeedLimit* = 70 km/h. Concerning *Area* and *District* dimensions, they are included in the hierarchy h_1 so that $Dim(h_1) = \{ALL, Area, District\}$ and $District \succeq_{h_1} Area \succeq_{h_1} ALL$ (*ALL* denotes the coarsest aggregation level and is always included in a hierarchy). Other examples of hierarchies are h_2 ($Dim(h_2) = \{ALL, Type\}$), h_3 ($Dim(h_3) = \{ALL, SpeedLimit\}$) and h_4 ($Dim(h_4) = \{ALL, MileageExtension\}$).

Anomalies and Metadata. Fig. 3 illustrates two streams of measures, referred to a road portion belonging to the facet $\hat{\phi}$ from the example above, collected after two different anomalous events have been recognised: at the top, the type of the anomaly is a depression, while in the second stream it is a hole. The anomalies have been detected by the black box of a vehicle with an average speed of 45 km/h (in case of depression) and an average speed of 53 km/h (in case of hole). The latter information constitutes metadata regarding the vehicle and is associated with the measurement session as well.

4. Clustering-based data summarisation

The streams of measures within a measurement session, collected on a specific road portion, are inspected to ascertain whether the road surface conditions are deteriorating or not. To obtain a synthetic representation of the temporal evolution of road surface conditions, data summarisation techniques based on an incremental clustering algorithm are applied [9]. Clustering offers a two-fold advantage: (a) it gives an overall view over measures, using a reduced amount of information; (b) it allows to monitor the conditions of the observed road portion better than single measures, that might be affected by noise and false outliers (e.g., due to sensing malfunctions) while observing a physical phenomenon of interest. The adopted algorithm relies on a lossless representation of a set of measures close each others, denoted as *synthesis*. A synthesis s_k, while observing a set of features on a specific facet, corresponds to a cluster of close measures of the observed features.

An incremental clustering algorithm applied on a stream of measures produces, at a given time t, a set of syntheses $\mathcal{S}(t)$, starting from: (a) the portion of the stream between the timestamp $t - \Delta t$ and the timestamp t, being Δt the temporal interval over which the set of syntheses is updated; (b) the previous set of syntheses $\mathcal{S}(t - \Delta t)$. Syntheses conceptually represent a specific status of the road portion, by observing a data stream. The set of syntheses $\mathcal{S}(t)$ composes a *snapshot*, i.e., a data structure which, for the smart mobility domain considered in this paper, has been defined as follows.

Definition 3 (*Snapshot*)**.** A snapshot for a road portion ρ, stored at time t considering a set of observed features $F \subseteq \mathcal{F}$, is defined as follows:

$$SN_{\langle F,\rho,\mathcal{M},\epsilon\rangle}(t) = \mathcal{S}(t) \tag{1}$$

where: (i) $\mathcal{S}(t)$ is the set of syntheses, identified at time t, based on the measures of features in $F \subseteq \mathcal{F}$ gathered on the road portion ρ; (ii) $\mathcal{M}$ is the set of metadata associated with the measurement session; (iii) ϵ is the event type (e.g., bump, hole) associated with the measurement session. When they are obvious, the set of features $F \subseteq \mathcal{F}$, the set of metadata $\mathcal{M}$ and the event type ϵ can be omitted and the snapshot is denoted with $SN_\rho(t)$. □

According to Definition 3, a snapshot refers to a specific road portion ρ, that in turn is associated with a facet ϕ_i. A snapshot is generated when the set of observed features and the set of metadata have been established. Intuitively, comparison between different snapshots is meaningful if the two snapshots share the same set of features and are generated starting from measures collected in the same conditions (i.e., with the same metadata). In this case, the two snapshots are defined as *comparable*. Fig. 4 shows the set of syntheses in four comparable snapshots taken at time $t_1 + k \cdot \Delta t$, $k = 0, \ldots 3$, where measured features are X–Y accelerations and the event type is *hole*. Specifically, they are referred to a road portion identified through a facet that groups urban roads in city downtown. In the figure, each red circle represents a synthesis: the centre of the circle is the centroid of the synthesis, whereas the radius reflects the dispersion of the data points (measures) in the synthesis, calculated as the RMS (Root Mean Square) deviation of the data points from the centroid. Fig. 4 displays a *sequence of snapshots*, that provides a vision of the evolution of road surface conditions over time. From Fig. 4(a) to (d) changes in the four snapshots can be identified. In fact, the set of syntheses moves towards higher values of X axis acceleration and lower values of Y axis acceleration. A sequence of snapshots is formalised as follows.

Definition 4 (*Sequence of Snapshots*)**.** A sequence of snapshots is defined as the following tuple:

$$S_\rho(t_1, \ldots, t_n) = \langle SN_\rho(t_1), \ldots, SN_\rho(t_n)\rangle \tag{2}$$

where $t_1, \ldots, t_n$ represent the time instants in which the snapshots have been computed on the road portion ρ, and $t_{k+1} = t_k + \Delta t$, $k = 1, \ldots, n-1$ hold. Snapshots in $S_\rho(t_1, \ldots, t_n)$ share the same event type, the same features and the same metadata.

4.1. Identification of relevant road portions

Given two comparable snapshots $SN_\rho(t_1)$ and $SN_\rho(t_2)$ (with $t_2 > t_1$), changes between syntheses in the two snapshots are apt to identify *relevant road portions*. Relevant road portions

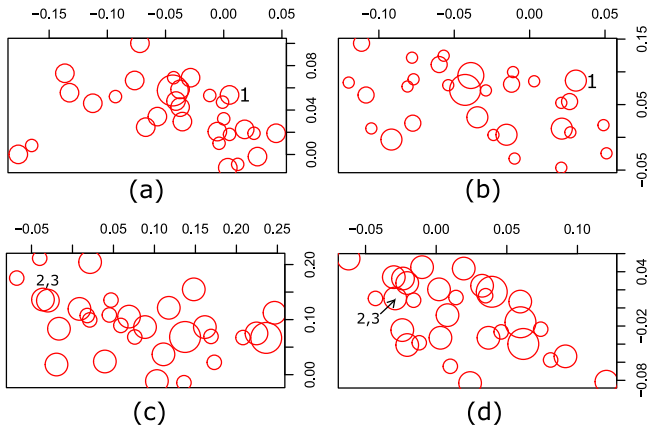

Fig. 4. Visual representation of a sequence of snapshots, resulting from the execution of the incremental clustering of a stream of records over X and Y axes acceleration: the sequence of four snapshots for a *hole* event is shown here.

are proposed to road maintainers to start the exploration from. In particular, the measure of relevance leverages the notion of *distance* between snapshots, formalised as follows.

Definition 5 (*Distance between Snapshots*)**.** The distance between two snapshots $SN_\rho(t_1)$ and $SN_\rho(t_2)$ on the same road portion ρ is based on the sets $\mathcal{S}(t_1)$ and $\mathcal{S}(t_2)$ of syntheses in the two snapshots (containing n and m syntheses, respectively, where n and m do not necessarily coincide) and is defined as follows:

$$d_{SN}(SN_\rho(t_1), SN_\rho(t_2)) = \frac{\sum_{s_i \in \mathcal{S}(t_1)} d(s_i, \mathcal{S}(t_2)) + \sum_{s_j \in \mathcal{S}(t_2)} d(\mathcal{S}(t_1), s_j)}{m+n} \tag{3}$$

where $d(s_i, \mathcal{S}(t_2)) = min_{j=1,\dots m} d_s(s_i, s_j)$ is the minimum distance between the synthesis $s_i \in \mathcal{S}(t_1)$ and syntheses $s_j \in \mathcal{S}(t_2)$, $\forall j = 1, \dots m$. Similarly, $d(\mathcal{S}(t_1), s_j) = min_{i=1,\dots n} d_s(s_i, s_j)$. To compute the distance between two syntheses $d_s(s_i, s_j)$, different factors have been combined: (i) the Euclidean distance between syntheses centroids, to verify if s_j moved with respect to s_i; (ii) the difference between syntheses radii, to verify if there has been an expansion or a contraction of s_j with respect to s_i; (iii) the difference in the density of syntheses (i.e., number of aggregated records with respect to the hyper-volume occupied by each synthesis).

Apart from the qualitative temporal evolution of the syntheses in a sequence of snapshots $S_\rho(t_1, \dots, t_n)$ (in brief, S_ρ), that can be visualised as depicted in Fig. 4, a quantitative assessment of the evolution of syntheses in S_ρ can be computed based on Definition 5. Since the collection of a data stream and corresponding calculation of syntheses and sequence of snapshots is triggered by the occurrence of an anomalous event, such a quantitative assessment will be used to monitor the evolution over time of the anomaly on the road portion identified as relevant. The quantitative assessment of the evolution of snapshots in a sequence S_ρ is based on a vector of distances, denoted with $\mathbf{d}_{S_\rho}$. In particular, each component of $\mathbf{d}_{S_\rho}$ is calculated as:

$$\mathbf{d}_{S_\rho}[i] = \frac{d_{SN}(SN_\rho(t_1), SN_\rho(t_{i+1}))}{\Delta t} \; \forall i = 1, \dots, n-1 \tag{4}$$

where: (i) $SN_\rho(t_1)$ is the first snapshot of the sequence, fostered as a reference snapshot to perceive the temporal evolution of the anomalous event from the beginning of the sequence; (ii) Δt is the syntheses update temporal interval, used as a scaling factor to balance each distance value. The distance vector $\mathbf{d}_{S_\rho}$ can be exploited to establish whether a sequence of snapshots has to be considered as *relevant* or not, according to the following definition.

Definition 6 (*Relevant Sequence of Snapshots*)**.** Let $\bar{d}_{S_\rho}$ be the average value of all the distances contained in $\mathbf{d}_{S_\rho}$. A sequence of snapshots S_ρ is relevant if:

$$|\{\mathbf{d}_{S_\rho}[i] \; such \; that \; \mathbf{d}_{S_\rho}[i] > \bar{d}_{S_\rho}\}| > k \cdot dim(\mathbf{d}_{S_\rho}) \quad (i = 1, \dots, dim(\mathbf{d}_{S_\rho})) \tag{5}$$

where: (i) $dim(\mathbf{d}_{S_\rho}) = |S_\rho| - 1$ is the number of components of $\mathbf{d}_{S_\rho}$, being $|S_\rho|$ the number of snapshots contained in the sequence; (ii) $\mathbf{d}_{S_\rho}[i]$ denotes the ith component of $\mathbf{d}_{S_\rho}$; (iii) $k \in [0, 1)$ is a parameter used to set the sensitivity of relevance evaluation. The higher k, the lower the probability to identify the sequence of snapshots as relevant. For example, $k = 0.5$ means that a sequence of snapshots is recognised as relevant if at least 50% of distances in the vector $\mathbf{d}_{S_\rho}$ is greater than the average value of distances in the vector. If a sequence S_ρ is identified as relevant, then the corresponding road portion ρ is considered as relevant as well. □

5. Relevance-based data exploration methodology

Clustering-based data summarisation and the measure of relevance are at the basis of the definition of a methodological approach for mobility data exploration. Specifically, clustering-based data summarisation reduces the volume of target data, focusing on aggregated data representation (syntheses) instead of single measures, that might be affected by high variability influenced by noise and perturbations in the sensing infrastructure. The measure of relevance is used to quickly identify relevant road portions, guiding the exploration towards deteriorating situations only, thus enabling road maintainers to perform data inspection in presence of a large number of road portions. By relying on the Multi-Dimensional Model, the road maintainers can move across relevant road portions previously identified.

5.1. Exploration scenarios

Road maintainers may focus the exploration on different road portions, depending on their exploration goals. As a result, a set of exploration use cases, that we refer to as *exploration scenarios*, can be performed. An exploration scenario aims at capturing the essential elements demanded to perform data exploration on relevant sequences of snapshots according to Definition 6, in order to fulfil a specific exploration goal, amongst the ones considered in the MoSoRe project. Exploration scenarios are formally defined in the following.

Definition 7 (*Exploration Scenario*)**.** An exploration scenario ES_i is a triple $\langle goal_i, \phi_i, \sigma_i \rangle$ where: (i) $goal_i$ is a textual description of the goal of the scenario; (ii) $\phi_i \in \Phi$ is a facet to filter the road portions involved in the scenario; (iii) σ_i is a sorting function to be applied on the road portions classified with ϕ_i, depending on $goal_i$, for their inspection by the road maintainer. □

In the following paragraphs, three exploration scenario with different goals, employed in the smart mobility context, are described.

ES_1 **-Prioritisation of anomalous events of the same type.** In this exploration scenario, sequences of snapshots related to the same type of anomalous event (e.g., hole) are considered. In particular, the goal is to enable road maintainers to choose among

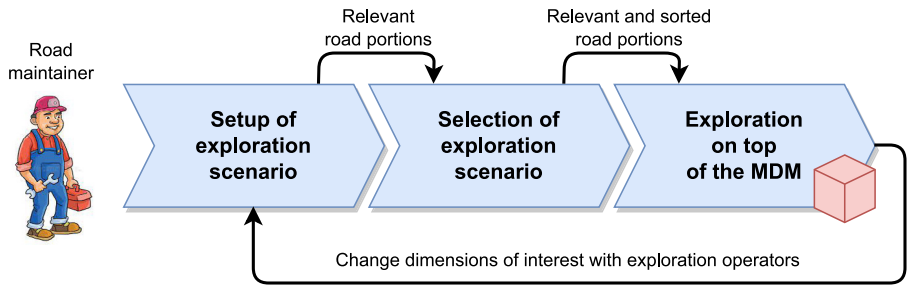

Fig. 5. Overview of the methodological phases for data exploration.

different road portions on which the same type of anomalous event has been detected. Let $S_{\hat{\rho}_1}$ and $S_{\hat{\rho}_2}$ be two sequences of snapshots summarising streams generated on the occurrence of two anomalous events of the same type. Definition 5 is used to establish whether the event corresponding to $S_{\hat{\rho}_1}$ has to be considered more critical than (or, equivalently, having the highest priority with respect to) the one corresponding to $S_{\hat{\rho}_2}$. To this aim, two vectors of distances $\mathbf{d}_{S_{\hat{\rho}_1}}$ (for $S_{\hat{\rho}_1}$) and $\mathbf{d}_{S_{\hat{\rho}_2}}$ (for $S_{\hat{\rho}_2}$) are calculated. Therefore, the anomalous event corresponding to $S_{\hat{\rho}_1}$ is considered as more critical than the one corresponding to $S_{\hat{\rho}_2}$ if the mean distance value $\overline{d}_{S_{\hat{\rho}_1}}$ is higher than the mean distance value $\overline{d}_{S_{\hat{\rho}_2}}$. In this scenario, the sorting function σ_1 is leveraged to establish a priority order from the most to the least critical road portion, induced by the priority of events occurred within. The above mentioned criterion can be generalised when more than two sequences of snapshots are considered.

ES_2 -**Recurrent anomalous event of a given type.** In this exploration scenario, the number of relevant sequences of snapshots are considered. The goal of this scenario is to enable road maintainers to choose among road portions based on the frequency of anomalous events occurring on them. To this aim, for each road portion filtered through the facet ϕ_2 and for each known type of anomalous event, a counting operation is performed on the set of available relevant sequences of snapshots. Relevant sequences of snapshots are detected by applying Definition 6. As a result, it is possible to establish the type of the most recurring event on a specific road portion $\hat{\rho}$ and, generally, for all the road portions classified with ϕ_2. In this scenario, the sorting function σ_2 can be used to sort the road portions starting from the one with the highest number of relevant sequences of snapshots corresponding to a specific event type $\tilde{\epsilon}$.

ES_3 -**Recurrent anomalous event of a generic type.** A variant of the previous exploration scenario is not bounded to a specific event type $\tilde{\epsilon}$, but considers all available event types (for instance, in MoSoRe project, bump, hole, depression and rough ground).

The (non exhaustive) list of exploration scenarios just discussed is characterised by a common procedure to perform the exploration: relevant road portions are first selected and filtered through the facet ϕ_i and then sorted according to a specific criterion, being the seriousness of a given event type (ES_1), the frequency of a specific event type (ES_2) or the frequency of a generic event on the road portion (ES_3). Scenarios can also be composed. For instance, road maintainers can start exploration based on frequency of event types (ES_2 or ES_3) and then refine their search based on seriousness of a specific event type (ES_1). This suggests the definition of a methodology for inspecting relevant road portions, based on the scenarios, as explained in the following.

5.2. Exploration methodology

We conceive the mobility data exploration problem as the exploration of the relevant sequences of snapshots associated with the anomalous events, according to a methodology organised over the three phases illustrated in Fig. 5: (i) setup of exploration scenario; (ii) selection of exploration scenario; (iii) exploration on the top of the Multi-Dimensional Model. The three phases may be repeated as long as road maintainers: (a) move within the Multi-Dimensional Model, to restrict/broaden the set of road portions, using facets for exploration; (b) revise their exploration strategy, thus changing the scenario (goal).

1 – Setup of exploration scenario. In this phase, the road maintainer may specify his/her preferred dimensions in a facet ϕ_i, in order to delimit an initial set of road portions to start the exploration from. The expected output of this phase is a set of relevant road portions, detected within a given facet ϕ_i according to Definitions 5 and 6. The next two phases are aimed at shifting the attention of the road maintainer on the most promising road portions to explore (determined depending on the relevant sequences of snapshots).

2 – Selection of exploration scenario. After the exploration setup, the road maintainer may perform one of the exploration scenarios $ES_{1,2,3}$ presented in Section 5.1. The choice of the exploration scenario induces an order over the set of relevant road portions filtered through the facet ϕ_i, as relevant road portions are suggested to the road maintainer for exploration according to the sorting functions $\sigma_{1,2,3}$.

3 – Exploration on the top of the Multi-Dimensional Model. Within the scope of a scenario, the road maintainer may apply *exploration operators*, to change the dimensions instances in the facet ϕ_i according to his/her exploration interests. Exploration operators are inspired by the well-known OLAP operators (slice, dice, roll-up, drill-down) and are formalised according to the following definition.

Definition 8 (*Exploration Operator*)**.** An exploration operator is denoted as $o(\tau_o, \iota_o, d^k_{h_j})$, where: (i) τ_o is the type of the operator; (ii) ι_o is a measure quantifying the extent of variation in the number of relevant road portions as a result of the application of o over the dimension $d^k_{h_j}$ in the hierarchy h_j. We denote with $\mathcal{O}$ the set of available operators. □

Exploration operators are leveraged to move over the levels of the hierarchy h_j, thus modifying instances in ϕ_i accordingly, or to select/remove an instance of the dimension $d^k_{h_j}$. A change in ϕ_i induces a variation on the number of available road portions and, by extension, also on the relevant ones according to Definition 6. For example, a "drill-down" operation may restrict the size of the set of relevant road portions. To assess this variation and focus the attention of the road maintainer only on facets containing relevant portions, the following measure is used:

$$\iota_o = \frac{|\Sigma^{rel}_{o(\phi_i)}|}{max(1, |\Sigma^{rel}_{\phi_i}|)} \tag{6}$$

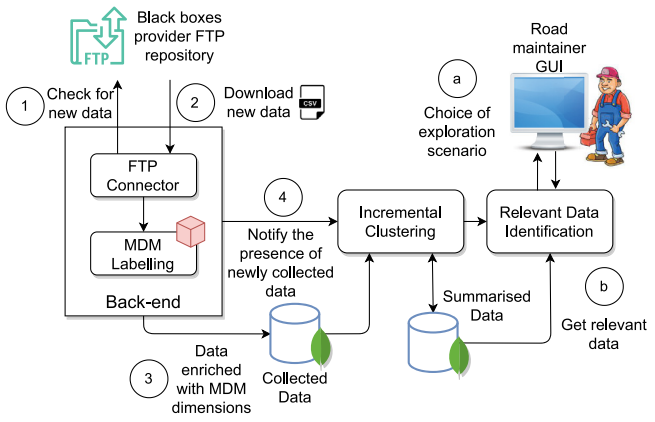

Fig. 6. Architecture overview.

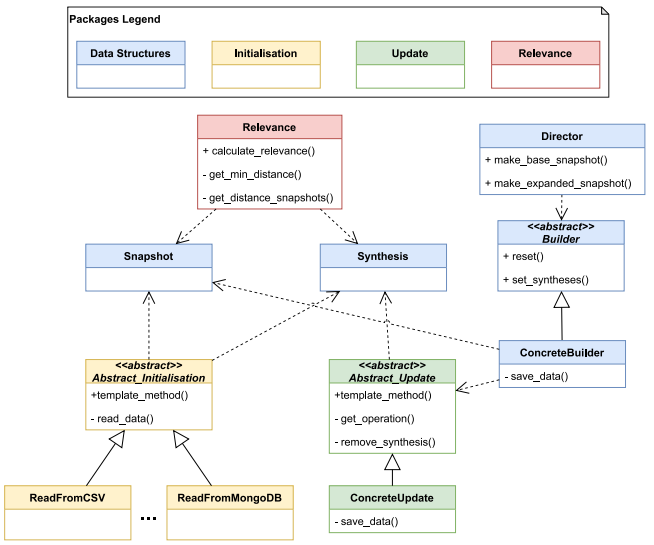

Fig. 7. UML diagram of the main packages of the Python software library that implements the Incremental Clustering and Relevant Data Identification modules.

where: (i) $|\Sigma^{rel}_{o(\phi_i)}|$ represents the number of relevant road portions after the application of o and (ii) $|\Sigma^{rel}_{\phi_i}|$ represents the number of relevant road portions before the application of o. A value of ι_o in [0, 1) denotes a reduction in the number of relevant road portions to be explored, a value >1 denotes an increment, whilst a value $= 1$ denotes no variations. Thus, the road maintainer may avoid a trial-and-error approach while moving across the dimensions of the Multi-Dimensional Model, leveraging the value ι_o to shift towards the most promising dimension instances, determining facets with relevant road portions to explore, revising his/her exploration choices within a scenario.

6. Implementation

6.1. Architecture

Fig. 6 reports the high-level architecture of the system employed for mobility data exploration. The numbers on the arrows denote the interaction flow between modules. Mobility data is made available by the provider's black boxes through an FTP repository, wherein black boxes periodically push collected measures (stored in the form of CSV files). The Back-end modules: (i) check for newly available measures using an FTP connector (steps 1 and 2); (ii) store the measures labelled with the dimensions of the MDM and metadata into a MongoDB database named Collected Data (step 3). Collected data is stored as JSON document organised into collections (a collection contains daily data). The Incremental Clustering module is notified about the presence of new available data to process from the Collected Data store (step 4). The output of the Incremental Clustering module is stored within the Summarised Data MongoDB database and then sent to the Relevance Data Identification module, which is in charge of: (a) identifying relevant road portions according to the selected exploration scenario; (b) sending relevant road portions to be displayed on a GUI. In the figure, (a) and (b) denote the exploration flow triggered by the road maintainer through the GUI.

6.2. Data summarisation and relevance evaluation library

The Incremental Clustering and Relevant Data Identification modules have been implemented in Python (version 3.10.1). In particular, a software library has been designed to implement the clustering-based data summarisation algorithm whose fundamentals have been presented in Section 4 (such algorithm has been called IDEAaS — which stands for Interactive Data Exploration As a Service) by relying on the renowned `Template` and `Builder` design patterns from [10]. `Template` is a behavioural pattern which enables to define the skeleton of an algorithm in a parent class. The sub-classes of the parent one may override specific methods of the algorithm, without altering its original structure. This pattern has been fostered to handle the creation and the update of syntheses within snapshots, in order to be flexible enough to ensure future extensions (for instance, to implement a different update mechanism for syntheses). `Builder` is a factory pattern allowing the generation of complex objects in a step-wise manner, preventing the proliferation of highly-specialised sub-classes. This pattern has been adopted to pursue the flexibility in the creation of snapshots. The library is organised into five packages: a simplified UML class diagram is illustrated in Fig. 7. In particular: (a) the `data_structures` package regards classes for modelling and handling syntheses and snapshots; (b) the `initialisation` and `update` packages group classes devoted to implement the incremental clustering algorithm and its macro-phases (i.e., syntheses creation and update); (c) the `relevance` package gathers classes for the calculation of the measure of relevance. The data summarisation algorithm has been designed according to two different modes: (i) *single* execution or (ii) *iterated* execution, which is apt to test different configurations of clustering, by varying one of its parameters at a time, as it will be explained in Section 7.2. Noteworthy, the pattern-based structure of the software library has been leveraged to include the implementation of additional incremental clustering algorithms from the literature. Such algorithms have been included in the library to prove that the clustering quality delivered by IDEAaS outperforms the clustering quality obtained when fostering other existing reference algorithms. A thorough discussion on the latter aspect will be provided in Section 7.3.

7. Experimental evaluation

Experiments have been performed using the data summarisation and relevance evaluation software library described in the previous section on a PC equipped with an Intel Core i5-3210M processor, CPU 2.50 GHz, 4 cores, 8 logical cores, RAM 8 GB. The whole library has been deployed as a containerised application into a Docker container (Docker version 20.10.17). Indeed, containerisation is exploited to pave the way to a scalability

assessment of the proposed approach with different hardware configurations. The experimental evaluation touches different aspects: (i) the effects of IDEAaS algorithm clustering parameters on data summarisation quality (Section 7.2); (ii) the comparison of IDEAaS algorithm against other reference incremental clustering algorithms in the literature (Section 7.3); (iii) the assessment of the processing time, to prove that data summarisation and relevance evaluation can be efficiently computed (Section 7.4); (iv) usability tests on the prototype GUI to execute the exploration scenarios (Section 7.5). Experimental evaluation has been performed on mobility data acquired in the MoSoRe research project. The characteristics of this data are reported in the following.

7.1. MoSoRe mobility data

In the MoSoRe project, four different types of anomalies have been considered, namely, hole, bump, depression and rough ground. To explore road portions affected by these kinds of anomalies, data has been collected through vehicles black boxes and stored on the FTP repository as CSV files. Four different types of CSV files have been considered (in each file the pattern `YYYY-MM-DD` denotes the reference date of the measures contained within the file):

- `VEM_YYYY-MM-DD_TRIP.csv` contains information of start and end of a trip performed by a vehicle; it is composed of 18 columns containing information of place, time and weather conditions for start and end place and the mileage extension of the trip (in km); the number of records (that corresponds to the number of the trips in a day) ranges, on average, from 30 to 80;
- `VEM_YYYY-MM-DD_TRIPDETAILS.csv` contains the intermediate positions of a vehicle during a trip, gathering also metadata about the direction of the vehicle, heading and speed; the number of columns of the file is 12 and average size ranges from 1000 to 3000 records per day;
- `VEM_YYYY-MM-DD_HOBU.csv` contains information of the anomalous events occurred within a day; the type of event, the intensity of the impact, sampling frequency associated with the accelerometric traces, number of samples are retained in the file; the file consists of 11 columns and average size goes from 300 to 500 records per day;
- `VEM_YYYY-MM-DD_ACC.csv` contains the accelerometric traces associated with anomalous events that occurred on the monitored road portions; along with X-Y-Z acceleration values, each record contains a reference to the ID of the anomalous event and the relative position of the sample in the trace; the file consists of 8 columns and average number of records ranges from $2 \cdot 10^6$ to $3 \cdot 10^6$ records.

7.2. Effects of clustering parameters on data summarisation

The configuration of the IDEAaS incremental clustering algorithm is rooted on four parameters: (i) the maximum number of syntheses m, to be created during the initialisation phase of the algorithm and maintained over time for the whole duration of the data stream processing; (ii) a factor p, concurring in the calculation of the so-called *maximal boundary*, which is exploited during the assignment of incoming data points to syntheses (if the data point does not lie within the boundary of the nearest synthesis, then a new synthesis is created); (iii) the size of the time window Δt, retaining data points of the stream collected from timestamp $t - \Delta t$ to timestamp t; (iv) a threshold τ, influencing the ageing mechanism of syntheses, used to label the least updated syntheses, which are candidate to be discarded. The first two parameters

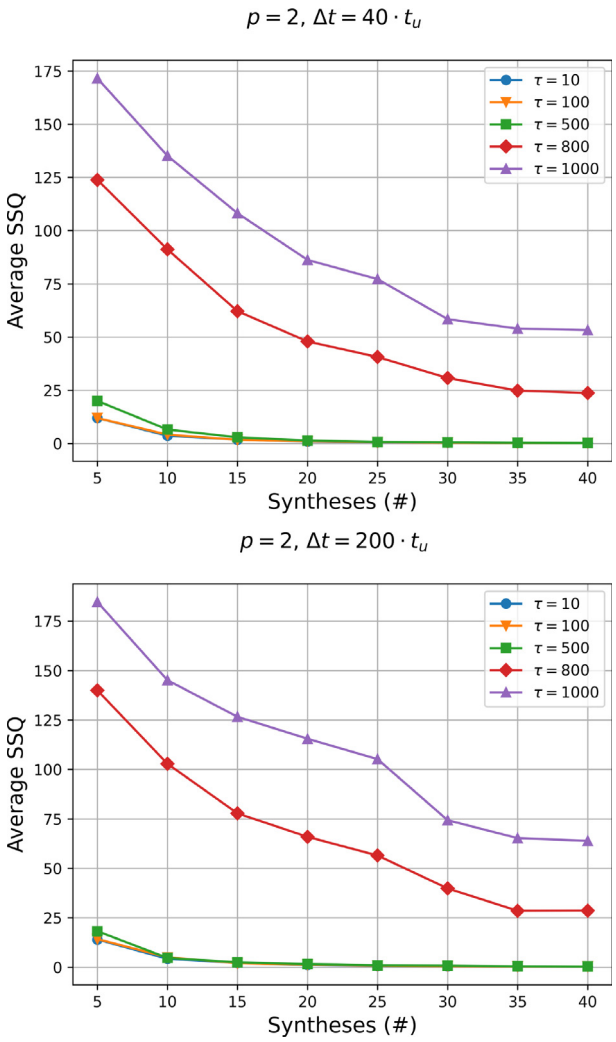

Fig. 8. Variations of the number of syntheses with different values of ageing threshold τ ($\Delta t = 40 \cdot t_u$ and $\Delta t = 200 \cdot t_u$).

can be analysed separately, in conjunction with the time window Δt and the threshold τ. In the following, we will describe tests conducted to demonstrate how the maximum number and maximal boundary of syntheses impact on data summarisation quality as applied on available mobility data, by varying at the same time the window Δt and the threshold τ. For all the tests, we considered a representative stream of measures denoting a hole anomalous event, composed of ≈2000 samples (Z axis acceleration only). For the evaluation of the quality of clustering, we relied on the renowned SSQ metric (sum of squared distance), used for validation of data stream clustering algorithms [11], which is calculated as follows. Consider the points collected during the last Δt interval. For each point, the centroid of its closest synthesis is found by computing the Euclidean distance. Hence, the SSQ at time t is equal to the sum of squared distances for all the points within the last Δt interval. SSQ is not bounded to any predefined range, and small SSQ values mean better compactness of syntheses (i.e., it quantifies the error due to the clustering operation). Therefore, SSQ must be minimised.

Maximum number of syntheses. We executed the IDEAaS clustering algorithm varying the parameter m (maximum number of syntheses) in the range [5, 40], with an increasing step of 5. In this experiment, we considered two values for the time window Δt, namely, $\Delta t = 40 \cdot t_u$ and $\Delta t = 200 \cdot t_u$ (where t_u is the minimum

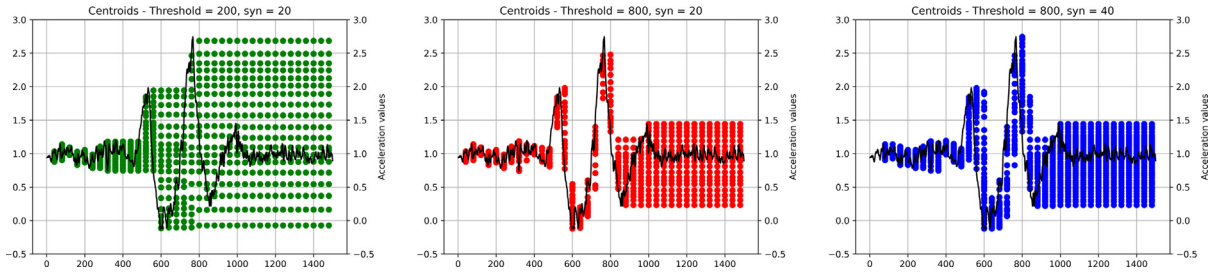

Fig. 9. Detail of centroids positioning with respect to original data (black line in the background) for different combinations of τ (ageing threshold) and m (number of syntheses); in the figure, $\Delta t = 40 \cdot t_u$ and the radii are not in scale.

time resolution of a single sample and, in the case of the reference stream, it is equal to 50 ms), in order to assess SSQ metric value on the same number of snapshots for all the experiments. The width of the time window has been chosen to produce snapshots with a medium-high frequency, thus allowing us to focus only on the other three parameters of the data summarisation algorithm.

Focusing on the ageing threshold τ, as it can be evidenced from Fig. 8, values of $\tau < 800$ yield lower values of SSQ (i.e., better clustering results). On the other hand, the value of time window Δt has a negligible impact on the average SSQ value. Lower values of Δt will increase the promptness in the identification of changes in data streams (by augmenting the frequency of controls), but at the same time has an impact on the response time of the algorithm, as demonstrated in Section 7.4. Indeed, since the threshold τ steers the ageing mechanism of syntheses, if we set τ to a value close to zero, old input data is hardly forgotten. Conversely, as the value of τ increases, the algorithm is forced to consider the most recent points only. This aspect clearly emerges in Fig. 9, which illustrates the distribution of centroids values and input data for $\Delta t = 40 \cdot t_u = 2$ s (that has been set to this value to permit, with a fine-grained level of detail, to qualitatively perceive the evolution of the distribution of centroids and corresponds to acceptable response times, see Section 7.4). Regardless of the value of τ, Fig. 8 shows that the average SSQ progressively decreases as the number of syntheses m increases, until it becomes stable (from the figure, after a number of syntheses equal to 20 for $\tau \in \{10, 100, 500\}$ and 35 for $\tau \geq 800$). The plots in Fig. 8 shed light on two considerations. The first one is that, to achieve high-quality clustering and keep small the amount of memory dedicated to syntheses, a trade off value of m must (and can) be identified. The second consideration is that, if the ageing threshold τ is progressively increased, to improve the SSQ metric, a higher value of m is required. The latter case is evident if focusing on the points range $600 \div 1000$ in Fig. 9. Generally speaking, the lowest average value of SSQ is obtained for values of τ near to zero. In fact, as the stream is being processed, it will be reached a situation in which the syntheses are distributed from the maximum to the minimum peak of the acceleration signal, as removal of old syntheses is disabled. Hence, as a result of the disabled removal mechanism, a synthesis very close to input data will always be found (see the leftmost plot in Fig. 9).

Maximal boundary of syntheses. The p factor concurs in the assignment of incoming data points to existing syntheses, and it is leveraged to establish whether a new synthesis must be created for an incoming data point or not. The value of p should be chosen small enough, so that it can successfully detect most of the points representing new syntheses or outliers. At the same time, it should not generate too many unpromising new syntheses or outliers. In the experiments from Fig. 10, we tested different configurations of $p \leq 2$ (in particular, $p \in \{0.01, 0.1, 1.5, 2\}$) to assess the impact of its variation on clustering quality, with $\Delta t = 200 \cdot t_u$. The recommended value of $p = 2$ ensures the lowest average value of SSQ when varying the maximum number of syntheses (with fixed τ). Similarly, the same situation holds when varying the maximum value for the ageing threshold (with fixed m).

Final considerations. From the former experiments, it emerges that the two parameters which influence the most the quality of clustering are the maximum number of syntheses m and the ageing threshold τ, which therefore may influence the results of the exploration scenarios execution. For the considered representative input data, a maximum number of syntheses $m \in [20, 40]$ ensures a good quality of clustering. For the same range of syntheses, an ageing threshold τ near to zero delivers a better quality, disregarding the ability of the algorithm to consider only more recent data. The plot in the upper side of Fig. 10 shows that a good choice of the value of threshold would be in the range $[500, 700]$. For completeness, we repeated the experiments performed in this section on three other streams regarding hole events on the same road portion. Such streams have an increasing length (i.e., the number of data points, which depends on the generation frequency of the black boxes as explained in Section 2.1) of 4000, 6800 and 14160 points (collected every 50 ms), respectively. In particular, the latter corresponds to the maximum length of a stream related to an anomalous event in the context of the MoSoRe project. Focusing on the m parameter, that can be used to regulate the memory dedicated to store the syntheses, experiments evidenced that for the stream with a length of 4000 points, a value of $m \in [25, 45]$ ensures a good quality of clustering and a satisfying average value of SSQ. The same happens for the stream with length 6800 points and $m \in [30, 55]$, and for the stream with length 14160 points and $m \in [40, 75]$. For all the three streams, Δt, τ and p have been set to a fixed value.

7.3. Comparison between incremental clustering algorithms

After the in-depth analysis on how the configuration parameters of IDEAaS clustering algorithm affect the quality of clustering, we compared the quality of IDEAaS against the quality of other incremental clustering algorithms from the literature. Given the plethora of existing data stream clustering algorithms (a survey with the genealogical tree of the most famous algorithms and the mutual features they share can be found in [15]), we focused our analysis on the following algorithms: BIRCH [13], DenStream [14] and D-Stream [12]. In particular, DenStream [14] shares with IDEAaS the origin, being both the algorithms derived from the CluStream algorithm [16]. BIRCH [13] is the baseline algorithm, from which all the distance-based incremental clustering algorithms have been derived (including CluStream). Beyond a comparison against other distance-based algorithms, we also considered the ancestor of the grid-based ones, D-Stream [12], as suggested in [15]. Similarly to the experimentation described in

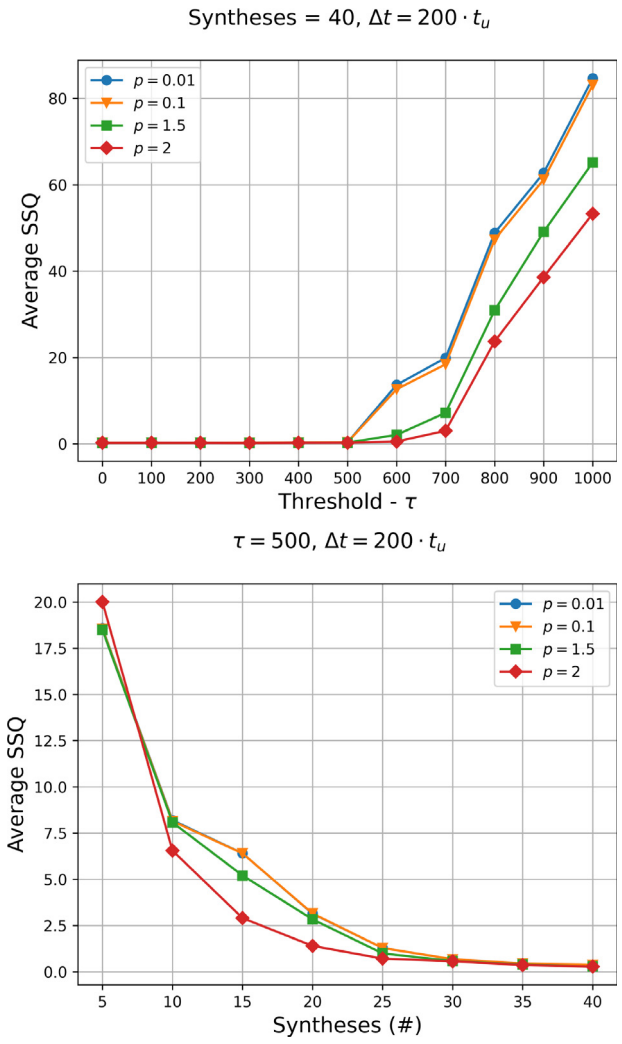

Fig. 10. Variations of ageing threshold for different values of maximal boundary factor (above, $m = 40$) and detail for $\tau = 500$ (below).

Table 1
Configuration parameters for the stream clustering algorithms (parameters not enlisted here have been set to their default values, provided in the corresponding paper).

Algorithm	Parameter	Value
IDEAaS	m	40
	τ	10 000
	p	2
D-Stream [12]	λ	0.9
	C_m	1.3
	C_l	1.1
	grid size	0.1
BIRCH [13]	*threshold*	0.1
	branching	6
	maxLeaf	20
DenStream [14]	λ	1.9
	ϵ	5
	β	1
	μ	1.5

the previous section, the clustering quality is expressed through the value of the SSQ measure. Comparison between the aforementioned representative clustering algorithms has been based on an accelerometric trail of ≈14k samples. The setup for the execution parameters of the clustering algorithms (reported in Table 1) has been obtained striving to meet two empirical criteria: (i) the stabilisation of the average SSQ, to identify a number of syntheses apt to contain the amount of memory used by the algorithms, similarly to the experiments in the former section; (ii) a proper setting for the parameters (where present) meant to regulate the capability of the algorithms to capture the dynamics of incoming data (e.g., the τ parameter of IDEAaS).

Fig. 11 illustrates the evolution of the SSQ quality index as the stream of data is processed by the clustering algorithms. For each clustering algorithm, the average value of SSQ (denoted as $\overline{SSQ}$) is reported in the legend in the top-left side of the plot. As can be evidenced from the figure, the quality of IDEAaS algorithm evaluated on the accelerometric trail data stream outperforms the quality delivered by the other reference clustering algorithms from the literature and, on average, IDEAaS assures the lowest SSQ value.

Final considerations and threats to validity. Apart from the quantitative extent of the clustering quality, the choice of a clustering algorithm with respect to another one is not a trivial task, since it has to consider a trade-off amongst several factors, which concur to delineate the advantages and disadvantages of clustering algorithms, as elicited by [8]. Two major factors are: (i) the degree of *domain knowledge* required for executing the algorithm, which is backed, for instance, by the setting of the configuration parameters (e.g., from Table 1, BIRCH and IDEAaS have less parameters with respect to DenStream and D-Stream); (ii) the underlying *structure of the algorithm* (e.g., even though DenStream shares with IDEAaS the same origin, the inner structure of DenStream is slightly more complex with respect to the structure of IDEAaS, since it introduces the management of core/outlier micro-clusters in the original implementation of CluStream, along with time decaying and damped window model). As anticipated in Section 7.2, regardless of the clustering algorithm, an universal issue regards the tuning of its configuration parameters, which goes beyond empirical criteria. Therefore, future efforts will be devoted to implement finer techniques to find an optimal tuning for the parameters demanded for the execution of the algorithms depending on the input dataset, resembling to the strategy proposed in [17]. Nevertheless, for the aforementioned considerations about the selection criteria to choose the algorithms to compare against IDEAaS, according to the classification described in [15], the selected ones are good representatives of the available solutions and the experimental results are successful in demonstrating the high clustering quality of IDEAaS.

7.4. Experiments on processing time and relevance evaluation quality

Regarding processing time, we performed tests varying the time window Δt. Fig. 12 shows the average time required to process a single record of measures for different Δt values, considering the reference stream of measures employed for the experiments described in Section 7.2. The figure shows that, on average, lower Δt values demand more time to process data. In fact, every time data summarisation and relevance evaluation are performed, some initialisation operations are executed (e.g., access to the set of syntheses previously computed). Therefore, to ensure lower processing time, the frequency of clustering execution and relevance evaluation must be reduced, that is, Δt value must be increased. On the other hand, higher Δt values indicate that clustering execution and relevance evaluation could be performed far from time instants where important variations occurred, thus reducing the quality of data relevance evaluation.

Therefore, we investigated the effects on the relevance measure when adopting different time windows Δt, jointly with variations of the ageing threshold τ. The goal of these experiments

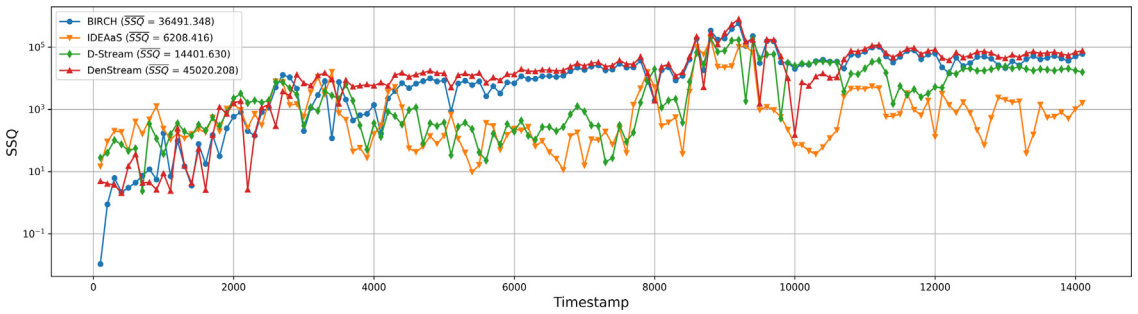

Fig. 11. Comparison of clustering quality against different stream algorithms (for all the algorithms, snapshots have been generated with $\Delta t = 100 \cdot t_u$).

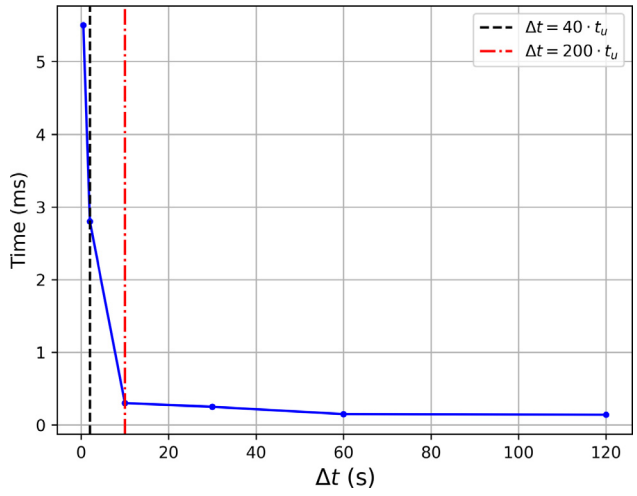

Fig. 12. Average processing time for each collected measure for different Δt values.

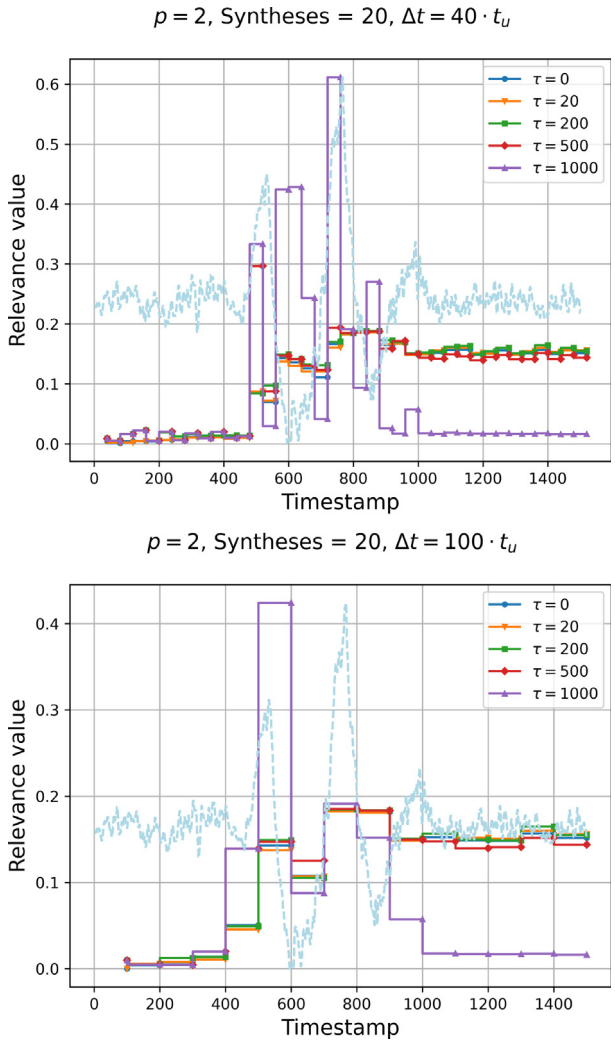

Fig. 13. Values of the measure of relevance for different ageing thresholds and time windows Δt (dashed line in the background is the original input data).

was to evaluate whether the distance-based metric, employed to identify relevant sequences of snapshots, is able to capture variations in collected data. To this aim, the first 100 measures in the reference stream have been employed to generate a snapshot, used as a reference to calculate the value of the metric for all the other snapshots that will be generated, according to Definitions 5 and 6. The value of τ permits to tune the sensitivity of the algorithm in following sharp variations of incoming data. Sensitivity is evaluated through the calculation of distance value. Indeed, a too small value of τ prevents old syntheses from being eliminated, and sharp variations are more difficult to be perceived. In Fig. 13 variations are hardly intercepted as τ value decreases. Additionally, there is a reduction in sensitivity when Δt assumes large values, as expected. Therefore, the rationale is to adaptively increase/decrease Δt according to the distance of relevant syntheses from warning and error thresholds for the observed features, depending on the road portion, since they correspond to potentially critical situations that must be monitored at finer granularity.

7.5. Proof of concepts GUI for mobility data exploration

A proof of concepts Graphical User Interface, that implements the proposed exploration scenarios, has been developed to support the exploration of mobility data. The GUI follows the methodological phases described in Section 5.2, namely, it enables road maintainers to start the exploration by selecting dimensions and their instances to form the facet ϕ (Phase 1), to select the exploration scenario (Phase 2) and to iteratively refine the exploration by changing road maintainer's choices on dimensions (Phase 3). At each step, the GUI proposes a set of relevant road portions and corresponding anomalous events. Road portions are ranked according to the sorting function σ of the scenario. As a representative example, in Fig. 14 the exploration scenario ES_1 (prioritisation of anomalous events of the same type) is considered. In the figure, the road maintainer, after selecting

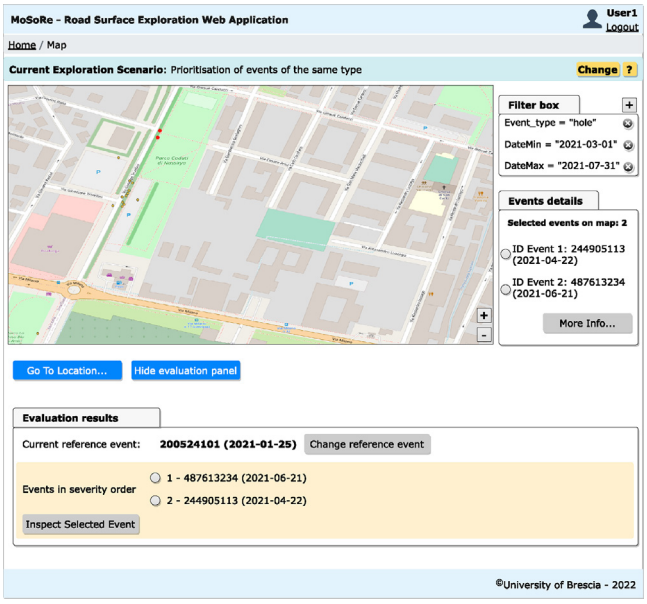

Fig. 14. Prototype GUI supporting the execution of exploration scenarios (detail of ES_1); the road maintainer, after selecting the exploration scenario, a road portion and an event type, is provided with the events of the chosen type, properly sorted according to their seriousness.

the exploration scenario, a road portion and an event type, is provided with the events of the chosen type, properly sorted according to their seriousness.

The GUI has been tested with usability experiments within the MoSoRe project. Usability experiments have been performed on a group of 25 users possessing road maintainers skills and familiar with software for viewing, editing, and analysing geospatial data (i.e., the so-called GIS — Geographic Information Systems). After an initial training, participants were assigned a task to be accomplished using the exploration tool, without imposing timeouts or any particular exploration constraint. Specifically, the task was a broad data exploration for inspecting and comparing events on a road portion with a high rate of anomalous events occurrence. Then, we asked participants to fill a standard System Usability Scale (SUS) questionnaire, which is widely employed for usability assessment purposes. SUS scores a software in a range between 0 to 100, where 0 indicates the least usability and 100 represents a high usability, respectively. For each statement of the questionnaire, road maintainers indicated their opinion on a five point scale. We averaged over all maintainers' questions and the resulting mean score amounted to 78, which locates our prototype tool in the 80–85 percentile range of the SUS score curve [35]. We also calculated the time participants spent to carry out their task. Using the prototype, participants accomplished their task in a shorter timer compared to a plain GIS interface without data exploration facilities based on the proposed scenarios.

8. Related work

This section surveys an excerpt of the existing Smart City platforms and frameworks, with focus on smart mobility issues (Table 2). In particular, our critical analysis focused on how research addresses big data exploration issues in this domain. One of the main challenges within Smart City environments regards the aggregation and processing of massive amounts of rapidly changing data, generated by sensors and IoT devices, in order to be ready for subsequent exploration. In this respect, the comparison between considered approaches is based on: (i) how *variety* of collected data within the Smart City domain is addressed; (ii) how *volume* of data has been tackled, and whether summarisation techniques have been employed to obtain meaningful data aggregations for analysis and exploration; (iii) the presence of *relevance evaluation* techniques to attract domain experts' attention while exploring data; (iv) if the approach takes into account an *exploration methodology* to ease exploration and (v) if an *implementation* environment has been provided to assess the approach.

Authors in [23] outline the requirements for a large-scale IoT data streams processing framework, addressing user's centric decision support and reliable information processing for Smart City applications. Therein, use cases involving traffic management and sustainable development of the urban fabric are presented, giving insights about the importance of exploration support. In [24], the CITIESData framework handles data streams from smart meters and sensors fostering a flexible ETL tool (called BigETL) apt to manage data with both different formats and with high generation rates. Through the framework, an evaluation on heating consumption data has been conducted. Similarly, to meet the rates of high-speed stream processing, an ecosystem based on Spark is proposed in [30], empowered by the adoption of graphics processing units (GPUs) to yield high elaboration performance. Focused on renewable energy systems analytics in Smart Cities, Preda et al. [29] adopts a methodology to process data coming from sensors installed on photovoltaic panels which envisages storage of data streams, resources management and a processing engine. Therein, three different analytical exploration scenarios are described, based on the application of Machine Learning techniques. Although not specifically focused on smart mobility, approaches in [24,29,30] have been analysed for their high-performance stream processing. Among them, only [29] proposes a data exploration solution. Grounded on addressing specific mobility problems (e.g., air pollution, traffic jam), the framework in [25] adheres to a Complex Event Processing (CEP) data-streaming processing paradigm and implements several real-time primitives to integrate, process and analyse the data streams. The work in [27] describes a model to implement an enterprise architecture, where several APIs are integrated, with the aim of analysing mobility data to pursue sustainable transportation. Authors from [28] propose a SOA (Service Oriented Architecture) to promote intelligent transportation, including proper services devoted to aggregate data and provide decision making support and data analytics, suggesting new perspectives for big data management in the presented Smart City domain. In [32], a platform for developing smart city applications has been proposed, to provide support for integration of heterogeneous data along with data analysis and visualisation. Therein, data analysis tasks are managed through a CEP engine and a Batch Processor. Focusing on energy consumption, the platform in [34] has been conceived as a modular system to promote a conscious use of energy by the users inside local energy communities. It includes a middleware apt to collect and analyse energy consumption big data, the latter achieved by means of dedicated services.

Further delving into the Smart Mobility topic, several research efforts propose the adoption of comprehensive solutions for big data analysis and exploration to improve mobility resilience. Authors in [18] propose a framework for analysing road accident data; therein, after data preprocessing, a clustering algorithm is applied and association rules are mined to obtain measures of interest, to find possible underlying patterns in the data set. With similar intents, the work in [19] combines IoT and big data to devise the Pavement Managements System (PMS), a road maintenance management structure composed of pavement detection

Table 2
Overview of surveyed big data exploration solutions.

	Focus	Big data exploration issues				Has implementation
		Addresses data variety	Addresses data volume	Relevance evaluation	Proposes exploration methodology	
Kumar et al. [18]	Analysis of road accident data	No	Yes (Clustering)	Yes (Association Rules)	No	Yes
Dong et al. [19]	Road maintenance	No	No	No	∼(Description of use cases)	Yes
Yang et al. [20]	Traffic state assessment	No	Yes (Clustering)	No	No	Yes
Alipour et al. [21]	Road maintenance	No	No	No	No	Yes
Zenkert et al. [22]	Analysis of road traffic and pollution	No	No	No	∼(Description of use cases)	Yes
Tönjes et al. [23]	Traffic management	No	No	No	∼(Description of use cases)	No
Liu et al. [24]	Analysis of heating consumption data	∼(Flexible ETL tool)	No	No	No	Yes
Junior et al. [25]	Analysis of traffic data	No	No	∼(Adoption of a CEP system)	No	Yes
Babar et al. [26]	Traffic congestion control	No	No (Only data aggregation techniques)	Yes (Rule engine with threshold limit values)	∼(Analytical use cases)	Yes
Anthony et al. [27]	Mobility and sustainable transportation	No	No	No	∼(Requirements for context-driven exploration)	No
Kemp et al. [28]	SOA to promote intelligent transportation	No	No (Only data aggregation services)	No	No	Yes
Preda et al. [29]	Renewable energies systems analytics	No	No	No	∼(Three different analytical scenarios)	Yes
Rathore et al. [30]	Big data stream processing	No	No	No	No	Yes
Bachechi et al. [31]	Big data analytics and visualisation for traffic monitoring	No	No	No (DTW for time series similarity assessment)	No (only visualisation layers built on top of queries)	Yes
Pereira et al. [32]	Big data integration and Smart City application development	No	No	No	No (only visualisation layers built on top of queries with user-defined filters)	Yes
Shir et al. [33]	Big data integration and Smart City application development	No	∼(hybrid scheme combining clustering, regression, classification)	No	No	∼(only experimental results reported)
Gagliardelli et al. [34]	Big data platform for analysis of energy consumption data	No	∼(no details on algorithms provided)	No	No	Yes
Ours	Road assessment	Multi-dimensional model	Yes (Incremental clustering)	Yes (Distance-based)	Yes (Formalisation of exploration scenarios)	Yes

and 3D modelling, data analysis and decision support. It also illustrates use cases for two main actors, the road maintenance company and a technical firm that offers smart solutions for road maintenance. In [20], a city traffic state assessment system is implemented using a big data cloud infrastructure, that hosts clustering methods to find areas of jam, thus assuring high scalability. Leveraging the recent advances in the field of computer vision and big data computing, authors in [21] developed a scalable framework for image-based monitoring of urban infrastructure, using both web images and Google Street View imagery to train a CNN model. Pursuing the goal of analysing road traffic and pollution data for the city of Aaruhs (Denmark), in [22] big data technologies ease the calculation and visualisation of the least polluted route. The approach by Babar et al. [26] aims at controlling traffic congestion by proposing a preliminary idea of analytical use cases. In their approach, data exploration is achieved by applying aggregation techniques, and a rule-based engine with thresholds (limits) suggests relevant data to end-users. Authors in [31] propose a framework to analyse urban traffic data through effective information visualisation techniques and the use of a dashboard. Data is collected from traffic sensors, and it is leveraged to build up a traffic simulation model. Focusing on user mobility (specifically, bike sharing ecosystems), in [33] a hybrid prediction scheme has been devised, to tackle renowned mobility challenges (e.g., prediction of the level of hourly demand of bikes).

Novel contributions. With respect to our approach, which considers the synergy between big data exploration techniques and methodological steps, the aforementioned approaches do not offer a comprehensive environment to perform big data exploration. This happens in the analysed Smart City platforms, providing only a high-level vision of how data exploration has been performed. For instance, approaches in [24,28,30] mainly focus on the design of the data stream processing architecture and on the efficiency of data acquisition algorithms, thus lacking of real analytic and exploratory use cases resembling our model of exploration scenarios. Moreover, methods and techniques targeted to face the variety and the volume of data are not envisaged in the exploration facilities offered to users, which are forced to rely only on a blind exploration strategy in the deluge of data. Both Tönjes et al. [23] and Anthony et al. [27] present examples of possible applications of the conceptual framework proposed in their work for exploration purposes, but it remains an abstract overview of requirements, not translated into methodological steps and quantitative metrics (e.g., the measure of relevance) to assist domain experts in the exploration of data streams as we propose in this paper. Similarly, Junior et al. [25] describe a case study where heterogeneous data streams coming from multiple sources are considered (e.g., bus fleet, city police force, city stations), but it is not framed into a methodology targeted to foster exploration. Even though exploration scenarios are introduced in [29], they are anchored to the application of specific Machine Learning algorithms, thus not prone to be generalised also for other domains, leaving in the background exploratory aspects. Pereira et al. [32] provide an abstraction layer for implementing dashboards for different categories of users (e.g., academia, citizens, industry, government), endowing users with the possibility of autonomously create queries and filters to limit data visualised on the underlying city map. However, the latter are not employed to set up exploration scenarios, as conceived in our proposal. In [34], use cases and scenarios are introduced, but they are strictly related to the description of dataflow operations (e.g., data sources management), thus not being targeted to end-user exploration.

We share with [18,26] the introduction of metrics/techniques to identify relevant data for road maintainers, in order to highlight the most interesting data summaries or to notify the occurrence of specific road network events. Nevertheless, multi-dimensional data organisation is not envisaged in any of these approaches. Furthermore, only [18,20,26] foster summarisation techniques, with [18,20] relying on clustering. Instead, the approaches in [31,33] employ different techniques, albeit not directly devoted to perform data summarisation. In particular, [31] adopts the Dynamic Time Warping (DTW) algorithm to analyse urban data flow time series, whilst [33] combines clustering, regression and classification. Authors in [26] apply a coarse-grained aggregation procedure, wherein data sources content is packed into blocks (recognised and classified according to their physical sources, such as sensors) and, within each block, aggregation is made by applying a divide-and-conquer approach on data sources attributes. In [18], several clustering algorithms from the literature are cited, but none of them is conceived to be applied incrementally on a data stream, whilst in [20] details on how the algorithm is applied are not provided. Regarding the formulation of exploration scenarios to support data exploration, only [22,26,31] sketch scenarios targeted to smart mobility, but details are coarsely given. Overall, in the surveyed smart mobility data exploration approaches, a methodology to steer data exploration is absent, thus forcing domain experts to adhere to a trial-and-error approach during exploration.

The approach described here is a substantial evolution with respect to its original formulation in [5]. In particular, the metric of relevance applied to the sequence of snapshots has been completely changed, eliminating the burden for road maintainers to define a reference snapshot for any possible kind of road portion and event type. This significantly makes easier the exploration tasks, thus not depending on road maintainers' experience. Exploration scenarios and experimental evaluation have been revised accordingly.

9. Concluding remarks

In this paper, we presented a methodological approach, that relies on big data exploration techniques, to support road maintainers during the inspection of road surface conditions in presence of multiple anomalies detected on the surface. The approach includes three components: (i) a multi-dimensional model, apt to represent the road network and to enable data exploration; (ii) a data summarisation algorithm, to simplify overall view over massive data streams collected by vehicles; (iii) a measure of relevance, aimed at focusing the attention of the road maintainers on portions of the road network that present critical conditions, thus enabling maintainers to properly plan interventions on the road infrastructure. The main contributions regard the introduction of a methodology for big data exploration, equipped with quantitative metrics to support road maintainers. The methodology has been declined into exploration scenarios, implemented in a prototype tool. The scenarios pave the way for the application of the methodology also in other application domains. Experimental results showed how relevance evaluation was able to efficiently attract the road maintainers' attention on road portions that present the most critical conditions and the proposed incremental clustering algorithm outperforms existing algorithms in the literature.

Future research directions are may-fold. On the one hand, a further enrichment of exploration scenarios will be pursued: (a) introducing personalisation aspects for users and the notion

of preferences in exploration; (b) expanding the amount of information gathered on the road surface. To this aim, in the MoSoRe project, on-board cameras are planned to be mounted on a group of representative vehicles, to be employed by road maintainers to collect video captures for the aforementioned purposes. On the other hand, a parallel implementation of the incremental clustering algorithm will be integrated. In fact, since some existing approaches [36,37] demonstrated how the CluStream algorithm (from which our incremental clustering algorithm has been derived) can be parallelised, in future work we will adapt the same parallelisation strategy to the algorithm proposed in this paper.

CRediT authorship contribution statement

Devis Bianchini: Conceptualization, Methodology, Writing – review & editing, Funding acquisition. **Valeria De Antonellis:** Conceptualization, Methodology, Writing – review & editing, Supervision, Resources. **Massimiliano Garda:** Software, Validation, Investigation, Data curation, Writing – original draft, Visualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgments

This study was carried out within the MOSORE project (Sustainable and Resilient Mobility - Project ID 1180965 - POR FESR 2014-2020) and within MICS (Made in Italy – Circular and Sustainable) Extended Partnership and received funding from Next-GenerationEU (Italian PNRR – M4 C2, Invest 1.3 – D.D. 1551.11-10-2022, PE00000004).

References

[1] C. Lim, K.J. Kim, P.P. Maglio, Smart cities with big data: Reference models, challenges, and considerations, Cities 82 (2018) 86–99.
[2] S. Paiva, M.A. Ahad, G. Tripathi, N. Feroz, G. Casalino, Enabling technologies for urban smart mobility: Recent trends, Opportun. Chall. Sens. 21 (2143) (2021).
[3] S.E. Bibri, The anatomy of the data-driven smart sustainable city: instrumentation, datafication, computerization and related applications, J. Big Data 6 (2019) 59.
[4] S. Campos-Cordobés, J. Del Ser, I. Laña, I.I. Olabarrieta, J. Sánchez-Cubillo, J.J. Sánchez-Medina, A.I. Torre-Bastida, Big data in road transport and mobility research, in: Intelligent Vehicles, 2018, pp. 175–205.
[5] D. Bianchini, V.De. Antonellis, M. Garda, Relevance-based big data exploration for smart road maintenance, in: 28th Int. Conference on Cooperative Information Systems (CoopIS 2022), 2022, pp. 19–36.
[6] A. Doan, Human-in-the-loop data analysis: A personal perspective, in: Proc. of ACM Int. Workshop on Human-in-the-Loop Data Analysis (HILDA'18), 2018, pp. 1–6.
[7] D. Tunkelang, Faceted search, Synth. Lect. Inf. Concepts Retr. Serv. 1 (2009) 1–80.
[8] A. Zubaroğlu, V. Atalay, Data stream clustering: a review, Artif. Intell. Rev. 54 (2021) 1201–1236.
[9] A. Bagozi, D. Bianchini, M. De Antonellis, A. Marini, A relevance-based approach for big data exploration, Future Gener. Comput. Syst. 101 (2019) 51–69.
[10] E. Gamma, R. Helm, R. Johnson, R.E. Johnson, J. Vlissides, et al., Design Patterns: Elements of Reusable Object-Oriented Software, Pearson Deutschland GmbH, 1995.
[11] S. Mansalis, E. Ntoutsi, N. Pelekis, Y. Theodoridis, An evaluation of data stream clustering algorithms, Stat. Anal. Data Min.: ASA Data Sci. J. 11 (2018) 167–187.
[12] Y. Chen, L. Tu, Density-based clustering for real-time stream data, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2007, pp. 133–142.
[13] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: A new data clustering algorithm and its applications, Data Min. Knowl. Discov. 1 (1997) 141–182.
[14] F. Cao, M. Estert, W. Qian, A. Zhou, Density-based clustering over an evolving data stream with noise, in: Proceedings of the 2006 SIAM International Conference on Data Mining, vol. 32, 2006, pp. 8–339.
[15] M. Carnein, H. Trautmann, Optimizing data stream representation: An extensive survey on stream clustering algorithms, Bus. Inf. Syst. Eng. 61 (2019) 277–297.
[16] C. Aggarwal, J. Han, J. Wang, P. Yu, A framework for clustering evolving data streams, in: Proc. of 29th International Conference on Very Large Data Bases (VLDB 2003), 2003, pp. 81–92.
[17] M. Carnein, D. Assenmacher, H. Trautmann, An empirical comparison of stream clustering algorithms, in: Proceedings of the Computing Frontiers Conference (CF 2017), 2017, pp. 361–366.
[18] S. Kumar, D. Toshniwal, A data mining framework to analyze road accident data, J. Big Data 2 (2015) 1–18.
[19] J. Dong, W. Meng, Y. Liu, J. Ti, A framework of pavement management system based on iot and big data, Adv. Eng. Inform. 47 (2021) 101226.
[20] C.T. Yang, S.T. Chen, Y.Z. Yan, The implementation of a cloud city traffic state assessment system using a novel big data architecture, Cluster Comput. 20 (2017) 1101–1121.
[21] M. Alipour, D.K. Harris, A big data analytics strategy for scalable urban infrastructure condition assessment using semi-supervised multi-transform self-training, J. Civ. Struct. Health Monit. 10 (2020) 313–332.
[22] J. Zenkert, M. Dornhofer, C. Weber, C. Ngoukam, M. Fathi, Big data analytics in smart mobility: Modeling and analysis of the aarhus smart city dataset, in: 2018 IEEE Industrial Cyber-Physical Systems, ICPS, 2018, pp. 363–368.
[23] P. Tönjes R. Barnaghi, M. Ali, A. Mileo, M. Hauswirth, F. Ganz, S. Ganea, B. Kjæ rgaard, D. Kuemper, S. Nechifor, et al., Real time iot stream processing and large-scale data analytics for smart city applications, in: European Conference on Networks and Communications, 2014, p. 10.
[24] X. Liu, A. Heller, P.S. Nielsen, CITIESData: a smart city data management framework, Knowl. Inf. Syst. 53 (2017) 699–722.
[25] M.R. Junior, R.P. de Oliveira, F. Carvalho, S. Lifschitz, M. Endler, M ensageria: A smart city framework for real-time analysis of traffic data streams, in: Workshop on Big Social Data and Urban Computing, 2018, pp. 59–73.
[26] M. Babar, F. Arif, Smart urban planning using big data analytics to contend with the interoperability in internet of things, Future Gener. Comput. Syst. 77 (2017) 65–76.
[27] B. Anthony, S.A. Petersen, A practice based exploration on electric mobility as a service in smart cities, in: European, Mediterranean, and Middle Eastern Conference on Information Systems, Springer, 2019, pp. 3–17.
[28] G. Kemp, P.L. Amaya, C.F. Da Silva, G. Vargas-Solar, P. Ghodous, C. Collet, Big data collections and services for building intelligent transport applications, Int. J. Electr. Bus. Manag. 14 (11) (2016).
[29] S. Preda, S.V. Oprea, A. others Bâra, Pv forecasting using support vector machine learning in a big data analytics context, Symmetry 10 (748) (2018).
[30] M.M. Rathore, H. Son, A. Ahmad, A. Paul, G. Jeon, Real-time big data stream processing using gpu with spark over hadoop ecosystem, Int. J. Parallel Program. 46 (2018) 630–646.
[31] C. Bachechi, L. Po, F. Rollo, Big data analytics and visualization in traffic monitoring, Big Data Res. 27 (2022) 100292.
[32] J. Pereira, T. Batista, E. Cavalcante, A. Souza, F. Lopes, N. Cacho, A platform for integrating heterogeneous data and developing smart city applications, Future Gener. Comput. Syst. 128 (2022) 552–566.
[33] B. Shir, J. Prakash Verma, P. Bhattacharya, Mobility prediction for uneven distribution of bikes in bike sharing systems, Concurr. Comput.: Pract. Exper. 35 (2023) e7465.
[34] L. Gagliardelli, L. Zecchini, L. Ferretti, D. Beneventano, G. Simonini, S. Bergamaschi, M. Orsini, L. Magnotta, E. Mescoli, A. Livaldi, et al., A big data platform exploiting auditable tokenization to promote good practices inside local energy communities, Future Gener. Comput. Syst. 141 (2023) 595–610.

[35] A. Bangor, P.T. Kortum, J.T. Miller, An empirical evaluation of the system usability scale, Int. J. Hum.–Comput. Interact. 24 (2008) 574–594.
[36] A. Bagozi, D. Bianchini, V. De Antonellis, Multi-level and relevance-based parallel clustering of massive data streams in smart manufacturing, Inform. Sci. 577 (2021) 805–823.
[37] P. Huang, X. Li, B. Yuan, A parallel gpu-based approach to clustering very fast data streams, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, 2015, pp. 23–32.

Devis Bianchini received his Ph.D. in Information Engineering in 2006 from the University of Brescia, where he is currently Full Professor in Computer Science Engineering and head of the Databases, Information Systems and Web research group in the Department of Information Engineering. His research interests include ontology-based resource discovery, service-oriented architectures, Big Data management and Web Information Systems design. He is chair of the Big&Open Data Laboratory at the University of Brescia. He is author of papers published in international journals and conference proceedings and he is referee for international journals. He coordinated national and regional research projects in the fields of Smart Cities and Industry 4.0.

Valeria De Antonellis is Emeritus Professor of Computer Science Engineering in the Department of Information Engineering at University of Brescia. Her research interests include advanced databases and web information systems conceptual modelling and design, conceptual schema matching and semantic integration, web resources semantic search and ranking. She participated in many European projects, among which INTEROP, RECITE II-DEAFIN, RENOIR, S-Cube, F3 and ITHACA. She authored numerous scientific publications, including articles, book chapters and books, she is member of the Steering Committee of the ER International Conference on Conceptual Modelling.

Massimiliano Garda achieved his Ph.D. in Information Engineering in 2017 at the University of Brescia, with a thesis concerning the design of data exploration techniques on top of (semantic) Data Lakes. Currently, he is a research fellow and a member of the Databases, Information Systems and Web research group in the Department of Information Engineering. Currently, he is investigating the use of Semantic Web methods and technologies for (Big) Data Exploration in several fields, including Smart Cities and Industry 4.0.