Smart Proximity: Annotating the Proximity of Entities In A Smart City Ontology

Hind A. Alawfi *haawfi@taibahu.edu.sa*
College of Computer Sciences and Engineering
Taibah University
Saudi Arabia

Khalid S. Aloufi *koufi@taibahu.edu.sa*
College of Computer Sciences and Engineering
Taibah University
Saudi Arabia

Abstract

The smart city concept contributes a new research area that will continue to be the focus of research for a long time. Different works have modelled and presented ontologies for smart cities, especially for data integration processes. In this context, obtaining a model in which the full functionalities of a DL reasoner are employed to generate new knowledge that would be available to the different devices in a smart city. This information can represent a useful picture of the environment around transports, hubs and people, enabling the smart devices in a city to make decisions according to this environment. We present a model of a smart city ontology with different axioms for generating new knowledge from available knowledge using a DL reasoner. This model considers the location and state of proximity between two entities in the environment. To implement our approach, we develop a tool referred to as smart proximity for generating and querying our smart city ontology. We expect the generated knowledge to be useful to many single working devices, especially devices that are available to transportation, and improve several functionalities such as motion, stop, waiting time and connections between two different means of transport.

Keywords: Smart City, Ontology, Proximity, Transportation, Reasoning, Decision Support.

1. INTRODUCTION

The world population is shifting into an increasing number of urban areas [1][2]. This shift has been projected to increase over time as the number of people securing jobs and other facilities in these cities increases [3][4]. The increased size of these cities raises a number of complex issues as cities need to manage transport and other infrastructures for improved delivery of services [5][6]. This management is critical to ensure that the air population, waste management, resources, health concerns and other resources, such as traffic congestion, can be managed in an ageing infrastructure. This approach will not only enable cities to improve over time but also ensure that they are able to grow in a sustainable manner [1], [2]. To address this changing nature of cities, numerous different ways of managing these expectations have been proposed. One approach is the smart city approach [5], [6]. A consistent definition of a smart city has not been suggested by academia or practitioners; however, it is consistently seen as a way to use new technologies to convert old cities into a smart initiative [7], [8]. All definitions agree that a smart city will be an urban space that is intended to enhance the everyday lives of citizens [9], [10].

A smart cities approach often attempts to use e-government, information systems, and technologies to integrate the issues faced by cities and obtain plausible explanations for addressing current issues [3][4]. In this context, the use of information and communication technologies (ICT) is considered a fundamental usage factor, which enables individuals and systems to communicate to improve the experiences, habits and facilities that are provided to individuals who live in urban spaces. Figure 1 illustrates the role of ICT in smart cities [11].

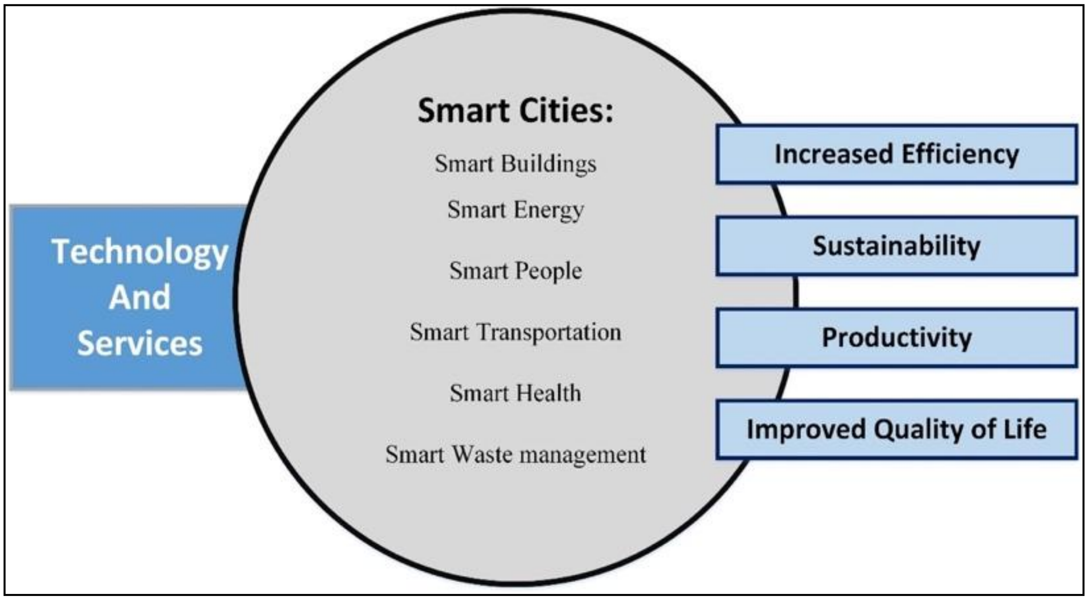

FIGURE 1: Role of ICT In Smart Cities [11].

A smart city is often seen as a forward-looking city, where smart combinations of various endowments of environment, mobility, governance, and economy are integrated to create a safe haven for citizens [7], [8]. The city is able to monitor and integrate the conditions to provide an ideal environment for various facilities [12], [13]. Critical infrastructure, such as transport, rail, and communication, can better optimize itself and use resources to undertake maintenance and other activities in a positive manner. Smart city resources can improve the way services are presented to citizens who live within a city. These services include infrastructure and energy management. In particular, improving transport management can be achieved using a smart city initiative since sensors and other information technologies help to organize the components of a transport system [14], [15].

Transportation in smart cities relies on numerous factors. In the first instance, the transport network needs to be integrated in a robust manner to ensure that efficacy of the various networks is maintained [16], [17]. This integration must be coupled with a series of innovations that can integrate various elements of the transport system and ensure that the system can learn from various mistakes that can be improved over time [17], [18].

A factor that is important to consider is that the transport network must be sustainable, which can reduce congestion and traffic [12], [13]. One approach that has been suggested for improvement in smart cities is the use of an integrated approach to ensure that various elements of the transport network can collaborate to ensure a long-term improvement in mitigating traffic and other problems in transport hubs [19]. The implementation of transport hubs and networks that collaborate in an integrated manner must be considered [18], [20]. The development of the transport hubs and networks must enable a citizen to obtain information and options on a continuous basis, which can ensure the sustainability of transport systems and improve the lives of citizens.

The importance of a transport system in a smart city is dependent on the ability of these transport networks to deliver sustainable change, which enables the delivery of new services and ensures that the various elements of a transport network can deliver long-term change and ensure that the sustainability of operations is managed [21]–[23]. However, these smart city initiatives can only be managed if a transport ontology is created, which can improve the semantics and structured data for ease of communication and coordination in smart cities. The goal of smart cities is to provide a more comfortable and safer environment that eases the daily lives of residents. Motivated by this goal, interest in developing more effective smart city applications and ontologies that deliver continuous improvement and address citizens and stakeholder's issues have increased.

One of the smart city areas that has a critical infrastructure with many issues is transportation. Different studies present transportation ontologies for smart cities; most of these studies are within data integration contexts. Most transportation ontologies in the literature focus on specific areas, such as traffic accidents, road traffic management, travel planning, and route planning. Transportation in smart cities can be improved, new knowledge can be produced from available information and smart city devices can facilitate decision-making based on this useful knowledge. The potential to improve the resource usage within the city, increase the effectiveness of the systems, improve the living standards and make smart cities smarter exists.

In this context, the usage of reasoning for generating new useful information is poor. Most previous studies do not consider the distance and proximity between two city objects. No smart city ontology attempts to store axioms for inferring knowledge starting with proximity information.

Different from other models, we take advantage of geospatial data that were furnished by different sensors to gain new knowledge using the reasoning abilities of a standard DL reasoner. With current technology, standard DL reasoners are not capable of inferring new knowledge using geospatial representations of points, lines, and shapes.

Our proposal is to annotate the proximity between two entities in the ontology for which the geospatial location is a known datum. Thus, the reasoner can infer new knowledge from the awareness that some entities are in the same vicinity. This new knowledge can be useful for smart devices in a city, with a focus on the aspect of smart transportation and increases their awareness of the characteristics of the environment in which they move and work.

Some general sample information that can be inferred from the entity proximity in cities is subsequently listed:

- No proximity between people and a bus stop means that the bus can avoid stopping near this bus stop.
- A person with a disability is near a street crossing: the person can have more time to cross the street.
- Ten cars near a parking zone with 10 spots means that the parking zone is probably full.
- Numerous people near a single transportation hub will require police intervention for managing the correct routing of people.
- When I select a destination on a transport, this destination can be translated to a near hub (e.g., “I want to go to the hospital”: a near transportation hub is selected as the destination, and then the person can travel by walking).
- A large truck near a street that is forbidden to trucks can be subjected to sanctions.
- People on a train can be informed whether taxis or shared cars exist near a next station.

We aim to answer the following questions:

- Can the proximity between two smart city objects be represented in an ontology?
- How can reasoning be utilized in ontologies to generate new knowledge?

- How can annotating proximity help smart city devices?

No smart city ontology can infer knowledge starting from proximity information. This task is difficult since geospatial data cannot be handled by standard reasoners.

We simulated the acquisition of several data from city sources by acquiring data about transportation from the open and available General Transit Feed Specification (GTFS) format of data [24] and simulating the acquisition of other data; all data were mapped using the same ontological representation. Due to the abilities of the OWL concept structure, we obtained all data using the same, unified collection of concepts. We subsequently exploited the reasoning abilities of a DL reasoner to obtain new knowledge from the raw data.

This paper is separated into several sections. After the Introduction, there is a section about related work followed by a section about research method describes the methodology for modelling the proximity ontology and annotating the proximity of entities. Then, a section follows about the results and discussion of implementing our smart proximity ontology are presented. The paper concludes with a section about conclusion to discuss the conclusions of our research and future work.

2. RELATED WORK

The importance of ontologies as a means for the representation of knowledge within a domain cannot be disregarded. Researchers often look for ways to identify the relationships between a domain and the concepts that define the domain. The transport community can benefit from defining new ontologies, as the data that are being generated from smart cities transport networks will not only be complex but also originate from a variety of sources. Transport data in an integrated smart city will often cause the development of the data from various sources, which can increase the complexity of the data. The complexity of these data will also be challenging, as different types of sensors, which are based on contradictory systems and settings, will create planning problems and delays.

Another problem with transport data in a smart city is the difficulty of mashing up data from various sources [21]. The semantics of the data and the sources of data, will often cause 'mash-ups' [21], which are not easily quantified. These mashups hinder the delivery, understanding, and reliability of the data. An ontology within the transport domain has numerous advantages, which enables a better conceptualization of specifications. Formal ontologies explicitly define semantics, which can be translated into machine-readable languages [21]. One advantage is that a formal ontology can support the different types of semantics and knowledge management issues that are faced by governments and other institutions, which enables the development of various options that can be addressed over time.

The use of these ontologies can produce formal methods, which can increase the sharing and exchange of information without the need for human intervention. These transport ontologies enable different types of systems in a vehicle and household to transmit information between two locations. These systems enable improvement in the data transfer rates, which facilitates the development of improved efficacy in transport networks.

Data management and ontologies need to be undertaken over time in a reusable manner, which enables critical development of large data. The handling of large volumes of data can support interoperability and challenge the different mechanisms that are needed for the smooth operation of smart cities. These transport ontologies have also been designed to scale in size as the system grows and create a system that can enable the development of operations. Numerous transport ontologies have been presented in the literature, which are examined in this review.

These transportation ontologies have been compared in Table 1 to establish the issues that must be addressed.

Name of Ontology	Proximity Concepts	Complexity of axioms	Evaluation	Knowledge Management Services
Content Personalization Transport Ontology [25]	No proximity concepts	Includes detailed axioms	No discussion of evaluation	For decision support
Ontology for Transportation Networks [22]	No proximity concepts	Beyond basic taxonomy	No evaluation defined	No services defined
Road Traffic Management Ontology [26]	No proximity concepts	Some axioms beyond subclass	Scenario-based evaluation is presented	Decision support
Road Accident Ontology [21]	No proximity concepts	No axioms beyond taxonomy	No evaluation defined	No services defined
Osmonto [27]	No proximity concepts	Concepts as taxonomy with no additional axioms	No formal evaluation	Route-finding via Open Street Map data
GenCLOn [28]	No proximity concepts	Concepts as taxonomy with additional axioms	Compared with model and case studies	Only query answering

TABLE 1: Comparison of The Discussed Transportation Ontologies.

Smart cities are increasingly dependent on the ability of different elements to communicate [23], [29]. Smart cities will have the ability to be monitored by ICT, which enables them to use data from a variety of sources in an integrated manner. The vision of a smart city relies on its ability to transform the lives and mobility of individuals, which can improve the administration [30], [31]. The big data challenges in the smart cities depend on the capability of smart cities to not only monitor the different data sets but also use them in an effective manner [23], [29].

Researchers must study the diverse aspects of a smart city, which can enable a degree of representation to be measured over the use of data. Diverse smart city aspects exist in the development of smart applications [32], [33]. The use of ICT should be considered, as the proximity between two different elements needs to be attained to ensure the long-term success of these projects [34]. The proximity of the objects in smart cities enables numerous advantages [34]. In the first instance, the use of ICT is critical for developing the technologies of smart city applications, which can use the proximity of objects to ensure that applications of smart cities can be implemented. Sensors impaled in vehicles and other objects must be employed to ensure that different location-aware systems can be understood and implemented [35], [36].

Numerous studies have examined the way in which different aspects of a smart city collaborate to not only develop and sustain improvement but also ensure that the development of new sensors can be utilized. The lack of practical knowledge about smart cities can also be disconcerting, as methods in which the proximity in transport networks can be meaningfully understood are need to improve the management of resources [14], [15]. For example, as shown in Table 2, many of the applications for the proximity in transportation are examined, and the success of each application can be considered [14], [15].

Application	Target	Validity (Scale of 10)
Smart People	Improvement in qualification, Ability to improve performance	8
Smart Governance	Participation in decision-making	10
Smart Living	Health and transport facilities	6
Smart Economy	Innovation in city	8

TABLE 2: Smart City Ontology In Transport.

A smart city transport network will use the proximity information from all resources to ensure that a coordinated response is given to the different types of information that is presented in the smart city. This information needs to be delivered in the most robust manner to improve the efficacy of the information that is being collected [5]. This approach can have numerous advantages for the participants as they can deliver continuous change and delivery of the services in a robust manner [5], [37], which will ensure that the various sensors can improve the services and prompt long-term change [38], [39].

This review has been used to propose an ontology that can include proximity as a key element in addressing the distances and entities in a smart city. This ontology can be developed over time to ensure that the position and distance of an object can be evaluated, which can boost the efficacy and effectiveness of smart city initiatives. The ontology will fill the gap, as a smart city ontology for transportation that can store axioms and infer knowledge from the proximity between two smart city objects, is lacking. This ontology helps to define the locations of objects and improve the interaction among these objects.

3. RESEARCH METHOD

3.1 Proposed Approach of Smart Proximity

A smart proximity tool is developed for city administrations to make data available in several devices in a city, such as buses, trains and traffic signals. The tool is focused on the smart transportation aspects of a smart city. Figure 2 shows the architecture of the proposed approach for smart proximity.

In particular, the smart proximity tool generates an ontology with information that can produce new knowledge to be furnished to devices in a smart city. This new knowledge is generated starting with proximity information between two entities in the ontology. The new knowledge can be useful for obtaining a portrait of the neighborhood of a device that works in the city, which enable the device to make decisions according to the aspects of its neighborhood and enables the device to present new functionalities. We proceeded as follows:

- We acquired data about transportation in the city of San Francisco (California, USA) that were shared with the General Transit Feed Specification (GTFS) format of data, as these data are available online. The acquired data is used to populate the ontology.
- We modelled an ontology that is related to these data with different concepts regarding entities that move in a city and locations, such as transports, people and local hubs.
- We proceed by defining the format in which GTFS data are shared, and then we present how we mapped these data in an ontological format.
- We added some simulated data, for example, the positions of some pedestrians in the city, as pedestrians are not available in GTFS data. Thus, the information required by the ontology that was not available in the acquired GTFS data was simulated by randomly generating some individuals in a way that was consistent with the remaining data.
- For a proximity representation, the proximity between two entities is annotated using a proximity entity of the type Proximity.

An example of annotating the proximity between a bus and a bus stop of 3 cm is presented as follows:

Individual: bus1
Types: Bus
Individual: busStop1
Types: BusStop
Individual: p1
Types: Proximity
Facts: hasEntity bus1, hasEntity busStop1, hasDistance 3.0

- We proceeded by defining some sample axioms in the ontology that can generate new knowledge from the available knowledge; for example, identify a bus near a hub with no people in the vicinity, which is useful as a bus can become aware that it does not have to stop near this hub, as no people have to be picked up.
- We implemented a tool that is referred to as smart proximity to generate and query the ontology.

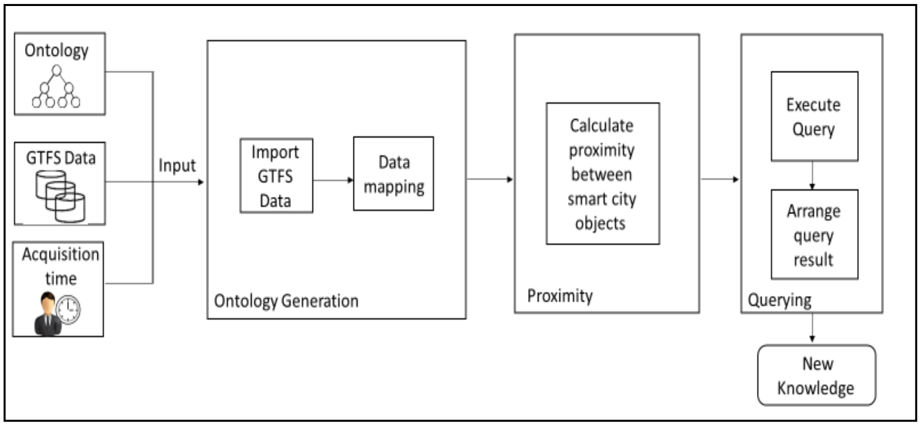

FIGURE 2: Smart Proximity Architecture.

3.2 Ontology Definition

We defined our smart city ontology by a data-driven approach. We start by identifying the concepts to model within the ontology. Although many concepts must be modelled in an ontology, we focused on the locations of transportation, people, hubs, and traffic signals, and the proximity between them since they are related to the acquired GTFS data. We build our ontology to enable other ontologies to reuse the entire ontology or some of its elements. Figure 3 illustrates the main classes and relations in our ontology.

The defined ontology consists of different concepts regarding entities that move in a city and locations, such as transports, traffic signals, people and local hubs. The following main classes are represented:

- Hub: is a single location in which people gather to enter or exist a mode of transport. A hub can be, e.g., a train station, bus stop, or taxi stand.
- Person: represents a single person walking in a city. We assume that a device exists in the city for tracking a person's location, such as available wearable devices or inference trough cameras.
- Transportation: represents the class of any transportation in the city. Bus, Cable Car, Rail and Taxi are subclasses of this class.
- Traffic Light: represents the set of all detected traffic signals in a city.
- Point: represents a point located on earth with a latitude and a longitude. A point is a concept imported from the WGS84 Geo ontology [40]. Every Hub, Person, Transportation or Traffic Signal is connected to a Point with the hasLocation property.
- Proximity: represents near objects in a city. Thus, each proximity individual represents two points that are located near each other.
- Disability: people can have one or more disabilities; these disabilities are annotated in the ontology. We assume that people with disabilities agree to anonymously share their status via a wearable device.

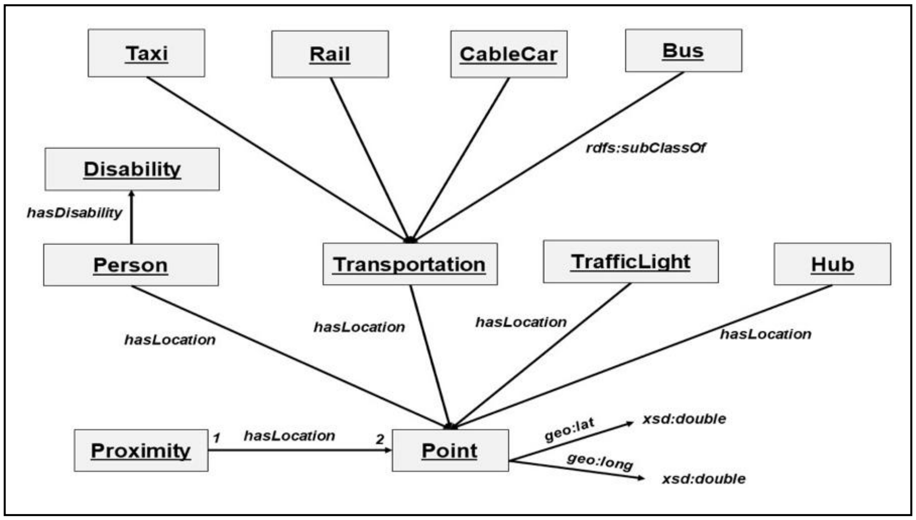

FIGURE 3: Main classes and Relations In Our Ontology.

3.3 Proximity Representation

In the smart city context, geospatial information is the core concept on which these smart cities are built. Geospatial information connects all smart cities objects, components, and applications. Thus, smart cities that are dependent on a geographical information system (GIS) due to their ability to efficiently model the real world of cities among other benefits aim to highlight the best features of smart cities [41]. The author in [41] argued that smart cities are spatially enabled cities, in which the spatial information of city objects are accessible by governments, stakeholders, and people.

Since everything in a smart city and its entities are based on geospatial information, we represent the proximity concept based on geographic coordinates, which are latitude and longitude. Consequently, our representation of proximity can be applied to other fields other than transportations with the goal of producing new knowledge and efficiently providing solutions. Our proposed tool can enable a user to import our representation of proximity into an external ontology. Thus, this tool can be applied to other smart city fields.

We realize that available DL reasoners cannot understand the proximity between two geo-located points, as the math that would be needed is not supported by current technology. Thus, in our proposed approach; the proximity between two Points is annotated in the ontology by generating an individual of the type Proximity and connecting two near points using the hasLocation property. Figure 4 shows an example of the proximity between two individuals of the type Person and Taxi.

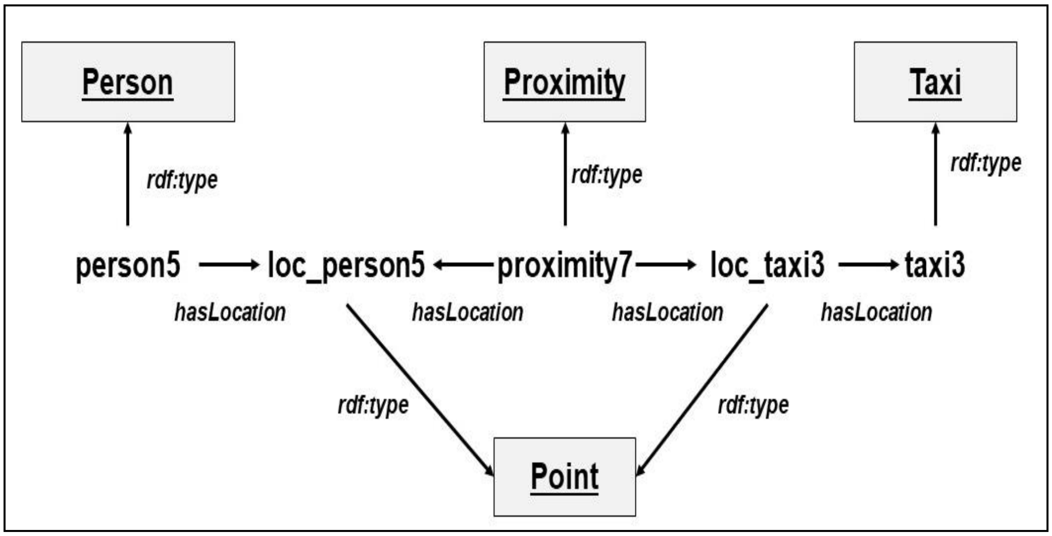

FIGURE 4: Example of proximity between two individuals of the type Person and Taxi.

We start by calculating the distance between two points. The distance between two points is calculated if their latitude and longitude are known. We compare the resulting distance with a pre-defined value that represents the maximum distance for two points to be considered near each other. If the distance is a pre-defined value, then we connect these two points with a proximity individual. Thus, all individuals of the proximity entity represent near points.

3.4 Algorithm of Mapping GTFS Data

To populate the ontology, we acquired data about transportation in San Francisco, which is one of the smart cities around the world. The data are presented in GTFS format, which is a standard format for sharing transportation schedules and other related information. The files that contain these data are Comma Separated Values (CSV) files. The files of GTFS data contain information about the trips performed by transportation and their times. They also contain information about the geospatial points associated with each trip and information about hubs locations for all trips. Additional details about the files of GTFS data are presented later. Our proposed tool offers the user the possibility to upload an external GTFS Data into our ontology.

We map the GTFS data into ontological propositions to be input into the ontology. Thus, the data are equivalent, but their format will change. The main scope of our ontology is to have at our disposal a single portrait of the locations of hubs, transportations, traffic signals and people in the city in a single instant in time. The locations of transportations in GTFS data are reported for the entire day. Thus, we do not link every information but only the information needed to identify the location of transportation in a single instant in time. Regarding transportation, we cannot directly acquire the location of the transportations as it is not reported in GTFS data for every instant. Thus, we must perform some estimations. We developed an algorithm for mapping GTFS data into the ontology. A pseudocode for this algorithm is presented in Figure 5. The algorithm proceeds as follows:

1. Select the time to perform the mapping. We refer to this time as the acquisition time.
2. Obtain all trips that are active during the selected time by checking if the acquisition time falls between the departure time and the arrival time of a trip.
3. Estimate the travelled distance by the transportation at the acquisition time by computing the time between the first departure time and the acquisition time. The distance is estimated by calculating the ratio between the time passed from departure and the total time of the trip. We use the same ratio to estimate the distance travelled at the acquisition time. For example, 10 minutes passed on a trip of 40 minutes; thus, 25% of the time of

the trip is passed. We estimate that 25% of the distance of the trip was travelled. We use this percentage to identify the point nearest to the calculated ratio. For example, the total distance of a trip is 2000 meters; 25% of 2000 is 500 meters. We look for a point for which the reported distance travelled to the nearest value is 500, that is, the point on which we locate the transportation. Thus, the latitude and longitude of the point are mapped in the ontology as the latitude and longitude of the transportation. Additional transportation information is acquired from the GTFS files.

4. Locations of hubs are mapped from the GFTFS files.
5. People are not available in the GTFS. Thus, we map them using a simulation. The locations of people are randomly generated in a city, with an improved probability in the proximity of a hub. For every four hubs, one hub has people.
6. Traffic signals are not available in the GTFS. Thus, we map them using a simulation. We create a file that contains traffic signal locations that exist in San Francisco. We map this file into the ontology.
7. Taxis are not available in the GTFS. Thus, we map them using a simulation. The locations of taxis are randomly generated in the city, with a better probability in the proximity of a hub. For every two rails, one taxi exists.
8. Every distance between two points is checked. Points with a distance lower than the pre-defined threshold are connected through a Proximity individual.

Algorithm MAPPING_OF_GTFS_DATA_PROCESS()

$acq = acquistion\ time\ selected\ by\ the\ user$
$T_o = set\ of\ trips\ in\ the\ trips.txt\ file$
$T_a = \emptyset$
$L = \emptyset$

for each $t \in T_o$

$$\textbf{do} \begin{cases} t_{dep} = first\ departure\ time\ of\ t\ in\ stops.txt \\ t_{arr} = last\ arrival\ time\ of\ t\ in\ stops.txt \\ \textbf{if}\ acq > t_{dep} \wedge acq < t_{arr} \\ \quad \textbf{do}\ \{ T_a \leftarrow t \end{cases}$$

for each $t \in T_a$

$$\textbf{do} \begin{cases} \tau = \frac{t_{arr} - acq}{t_{arr} - t_{dep}} \\ dist_{tot} = total\ distance\ of\ t's\ journey\ acquired\ from\ shapes.txt \\ dist_{acq} = \tau \cdot dist_{total} \\ t_{loc} = point\ with\ nearest\ distance\ to\ dist_{acq}\ from\ shape.txt \\ L \leftarrow (t, t_{loc}) \end{cases}$$

FIGURE 5: Pseudocode of Mapping GTFS Data Algorithm.

3.5 Reasoning Mechanism About Entities and Proximity

The ontology is considered a picture of a single instant in time of the location of people, hubs, transports and traffic signals in the city; hubs and traffic signals do not move, of course, and their locations are always identical each time. A location for each entity is annotated in the ontology with the proximity between two locations when their distance is below a certain threshold. We present four use cases for using this knowledge to generate new information that can be useful for the activities of decision-making of devices in a smart city environment.

Identifying hubs with people and hubs without people

Monitoring people at hubs is important for transport efficiency issues. Automatic transportation can only stop near hubs that have people. Taxis can efficiently move where they have a greater probability of being needed. Security can be efficiently moved where the risk of social issues is substantial.

If we want to identify hubs with more than 10 people, we can define a class **PopulousHub** as follows:

Class:	*PopulousHub*
SubClassOf:	*Hub and hasLocation some (Point and inverse(hasLocation) min 10 (Proximity and hasLocation some (Point and inverse(hasLocation) some Person)))*

which means a **Hub** in a point that is connected to a minimum of 10 **Proximity** individuals who are each connected to another **Point** that is the location of a **Person**.

Note that we need to set the assumption that all individuals are distinct to true in the ontology. Otherwise, the reasoner can assume that two or more individuals of the type **Person** refer to the same individual, which produces an unexpected inference behaviour.

We may want to infer the situation in which no people exist at the hub. Working the reasoner under the open world assumption, we are unable to make the reasoner automatically infer that no people exist at a hub. When we annotate zero people near the hub, the reasoner considers that some people that are not annotated in the ontology can exist near the hub due to the open world assumption status of the reasoner: If you do not state a proposition in the ontology, it may not be false.

During the mapping phase, we decided to annotate all hubs without people by enabling these hubs to be individuals of the class HubWithNoPeopleAround. Thus, hubs without people are marked during the mapping phase and not using reasoning.

Identifying transportation near hubs without people

When transportation approaches a hub, it usually stops; however, it does not have to stop if people do not exist at the hub. To identify transportation with empty hubs, assuming that HubWithNoPeopleAround is already computed in the ontology, we define the new class TransportationWithNoNearStop:

Class:	*TransportationWithNoNearStop*
SubClassOf:	*Transportation and (hasLocation some (Point and (inverse (hasLocation) some (Proximity and (hasLocation some (Point and (inverse(hasLocation) some HubWithNoPeopleAround)))))))*

which means a Transportation in a Point that is linked to a Proximity individual that is also connected to the Point, which is the location of a HubWithNoPeopleAround.

Triggering a slow stop status for a traffic signal as a person with a disability is near it

When a person with a disability needs to cross a street, providing this person with additional time to cross is reasonable.

This approach can be achieved by identifying traffic signals with people with some disabilities near them:

Class:	*SlowStopTrafficLight*
SubClassOf:	*TrafficLight and (hasLocation some (Point and (inverse (hasLocation) some (Proximity and (hasLocation some (Point and (inverse (hasLocation) some (Person and (hasDisability some Disability)))))))))*

Obtaining trains with the presence of at least one taxi at a nearby station
While riding the rails, a passenger may want to be informed using her/his wearable device, if taxis are available at the near station. We can identify trains that have at least one taxi in the proximity of the near station as follows:

Class:	*RailWithTaxiAtNearHub*
SubClassOf:	*Rail and (hasLocation some (Point and (inverse(hasLocation) some (Proximity and (hasLocation some (Point and (inverse (hasLocation) some (Hub and (hasLocation some (Point and (inverse(hasLocation) some (Proximity and (hasLocation some (Point and (inverse (hasLocation) some Taxi))))))))))))))))*

3.6 Querying the Ontology
In our proposed approach, we use SPARQL-DL queries to query the ontology. When executing a SPARQL_DL query; we display it result in a table. However, one of the problems of the SPARQL-DL queries is that the order of variables within the query expression is not retained when showing the results in a table. Thus, we must reorder the variables as we expressed them in the query.

First, we start by creating a table to record the results. Second, we obtain the order of the variables by extracting them from the query expression. The obtained variables are then written as columns of the created tables. We extract the rows of the query results by examining the results for the selected variable in their order. When we obtain them, we input the results in the correct cell of the row data. Last, we add the rows to the created table that has the correct order of variables as columns.

4. SMART PROXIMITY: RESULTS AND DISCUSSION
Smart proximity is a tool that is used to generate a transportation ontology, map GTFS data to the ontology and query it. Our tool provides four use cases that aim to provide new knowledge to smart city devices by allowing them to make decisions based on the entities near them.

We implement our smart proximity tool for generating the ontology and query it using JAVA 8. Our ontology is an OWL/RDF ontology that is defined via Protégé, which is an ontology editor for creating ontologies. Two APIs are employed: OWL API 3.4.4 and Java SPARQL-DL query tool. OWL API 3.4.4 is used to write the ontology and populate our ontology with GTFS data. Although SPARQL-DL API is used to query the ontology and obtain data from it. The JAVA SPARQL-DL query tool developed by the Derivo Company uses HermiT as a reasoner [42]. Figure 6 illustrates the main modules of the tool.

The implementation of our proposed approach undergoes five main phases—data acquisition, ontology definition, data mapping, proximity representation and query processing which are shown in Figure 6.

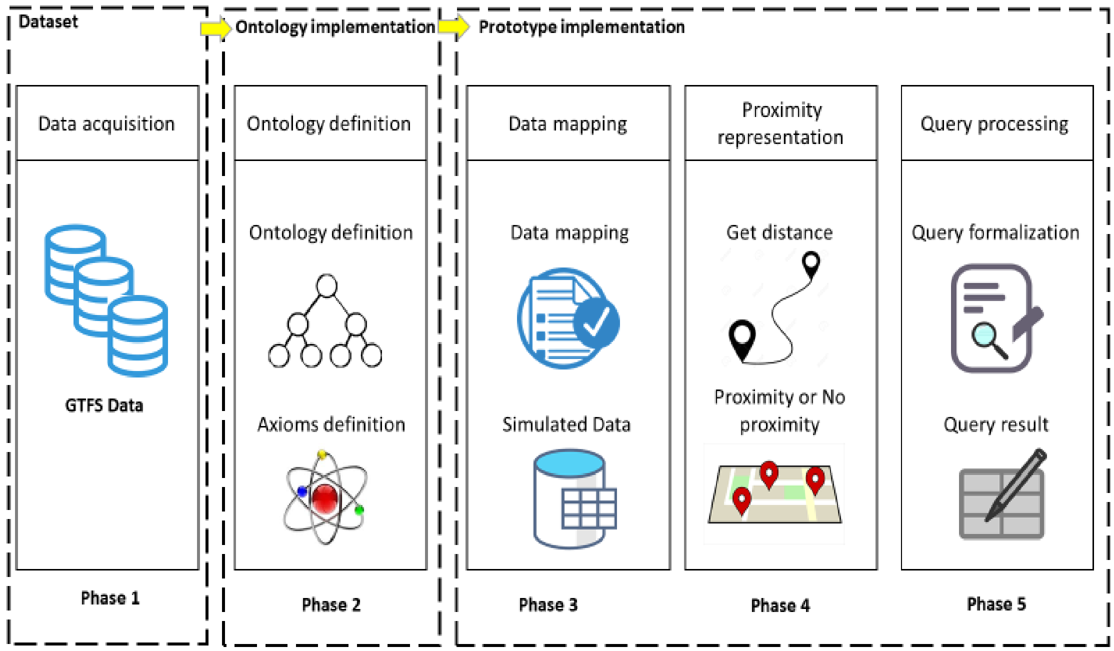

FIGURE 6: Main Phases of Implementing Smart Proximity Tool.

In what follows, we describe the implementation details of each phase of our proposed approach.

Data Acquisition

As stated earlier, we acquired data about transportation in the city of San Francisco (California, USA) shared using the format of GTFS data. Data available for the city of San Francisco are collected in several CSV files [24]. Data are collected into four files: Trips, Routes, Shapes, Stops and Stop times.

Smart Proximity Ontology

We start the ontology definition by defining the classes. We focus on the concepts that are identified and related to the GTFS data and our intended use cases. The defined entities are transportations, hubs, traffic signals, person, disability and proximity, which are defined as the main classes. Transportations include cable car, rail, bus and taxi as subclasses.

We import the point concept from an existing ontology that is referred to as the WGS84 Geo ontology, which provides the basic vocabulary for representing data with latitudes and longitudes [40]. The point concept is related to each entity and instance in our ontology.

We define two object properties in our ontology: hasDisability and hasLocation. The property hasDisability assigns a disability type to a disabled person, while hasLocation assigns a location of entities to the latitude and longitude values in our ontology.

We continue the ontology definition by defining two data properties: hasDistance and hasRouteName. The data property hasDistance allows double values to be assigned to individuals of proximity entity, while the hasRouteName property assigns all transportations with the route names of the trips that they conducted.

The resulting class hierarchy cannot provide the intended new knowledge for our use cases. Thus, we define axioms to acquire the new knowledge for the use cases. These axioms and the reasoning functionalities will produce new useful knowledge. We implement three use cases of our four use cases: Identifying transportations near hubs without people, identifying traffic signals with people with disabilities and Identifying trains with at least one taxi at a nearby station.

Smart Proximity Tool

The tool implementation consists of three main steps: Data processing and mapping, proximity representation and querying.

To perform the mapping, a user must select a time to populate the ontology with GTFS data. We obtain all trips that are active during the selected time. We identify the location of all transportations of the active trips according to the selected time. The location of the transportation is achieved by the setLocationAtGivenTime function.

After executing this function, all transportation will have a specific location with latitude and longitude values associated with it. We import the remaining files according to the selected time and randomly generate people and taxis. A sample code for adding random people around traffic signals is presented in this section.

As stated in section 3, our representation of the proximity starts by calculating the distance between two points, which is achieved via the getDistance function. The formula used in this function is obtained from GeoDataSource [43]. We check if the distance is below the threshold. If it is below the defined threshold, then add the individual of the type proximity.

Once the axioms presented are annotated in the ontology, the information that we need can be extracted from the ontology using a query mechanism that involves DL reasoning as a DL query or a SPARQL-DL query. We consider the latter technology.

After executing the query, we display it result in a table. Due to the SPARQL-DL problem of not following the order of the variables within the query expression, we reorder the result and display it in a table with its specified order by using the following code:

Moving to the user interface of the tool, we implement seven frames as interfaces to be shown to the user. Each frame performs a specific function, and we attempted to make them as user-friendly as possible. The interfaces capture the following tool procedures:

1. Selection of the ontology that contains concepts.
2. Selection of the GTFS data to be acquired.
3. Selection of a time and date for the acquisition of GTFS data.
4. GTFS data are acquired, other data is simulated, and an ontology is generated.
5. The user is asked if she/he wants to execute a query from a pre-defined list or a free query.

Ontology metrics:

Metrics	
Axiom	57685
Logical axiom count	43308
Declaration axioms count	14353
Class count	17
Object property count	3
Data property count	5
Individual count	14333

FIGURE 7: Ontology Metrics from Protégé.

The relative window is opened; when the user presses “Execute query”, the SPARQL-DL query is executed; and the results of the query are shown. To provide an overview of the number of classes, individuals, axioms and other metrics, we measure these metrics after populating the ontology with GTFS data. Figure 7 shows some of our ontology metrics.

To check the capability of our tool, some scenarios are executed to establish and conclude some important remarks for running the tool. The following scenarios are explained. We start by executing the four use cases with different times for mapping the data. The total measurements results are presented in Table 3.

Generating ontology parameters		Execution time			
Time	Date	Query 1	Query 2	Query 3	Query 4
1:00	2019/03/04	5 seconds	1 second (No result)	11 minutes	1 second (No result)
1:00	2019/10/15	3 seconds	1 second (No result)	11 minutes	1 second (No result)
1:00	2019/01/29	2 seconds	1 second (No result)	10 minuets	1 second (No result)
7:30	2019/03/04	3 seconds	3 seconds	14 minutes	18 minutes
7:30	2019/10/15	2 seconds	1 second	4 minutes	5 minutes

TABLE 3: Measurements of Queries Execution Time.

From Table 3, we conclude that the first and second queries require seconds to execute. The third and fourth queries require minutes in the case where they produce results. Thus, we can classify the third and fourth queries as long execution time queries. While we perform these measurements, we noticed that no result is returned from the query in some situations. No results mean that no proximity was observed among the entities specified in the query within the selected time. We also investigate the cases in which the ontology was generated. To successfully generate the ontology, we must ensure that the IRI of the ontology is correct and works. We must enter the time and date for generating the ontology in the correct format.

5. CONCLUSION AND FUTURE WORK

We present an ontology for mapping data and generating new knowledge from an existing ontology, which can be useful to smart devices in a city for understanding their surrounding environment and making decisions. This study is the first attempt to manage proximity information among entities in a city to generate new knowledge.

We demonstrated that new and useful information can be generated with the definition of proper OWL axioms in the ontology. We identified and mapped into an ontology the locations of different modes of transportation, which are represented in open GTFS data, in the city of San Francisco. We virtually represented the presence of people, taxis, and traffic signals in the city; this information is unavailable otherwise.

Addressing the proximity between two smart city objects was driven by the gap in the literature, where no smart city ontology has represented the proximity concept and generated new knowledge starting from this representation. This approach was also driven by the benefits and useful knowledge, which can be gained from the proximity awareness between two smart city objects.

Available knowledge is limited to knowledge that can be represented as a collection of OWL propositions in an ontology, while the potential knowledge that can be inferred is bound by the limits of expressivity of OWL axioms. The computational requests of a DL reasoner do not enable the generation of knowledge from many OWL axioms. However, we consider that this limit does not represent a problem for the usage of the approach in real-time environments.

A future challenge is to collect raw city data in real time and make it available to a smart device. Although we built a large ontology that represents an entire city in the current research, a device only needs information about its surrounding environment. Thus, the information that has to be made available to a device can be filtered a priori to include the minimal necessary knowledge in the picture of the environment, as represented by the OWL ontology. This minimal necessary knowledge is the representation of the individuals who are near the considered device. The representation of a device's surroundings usually does not consist of a large amount of data: entities near a single device can vary from 0 to even 1000; however, these are amounts can be easily managed by a DL reasoner, as we learned from our experiments. Thus, a device can use both the available (and always updatable) conceptualization and the minimal received information to generate new knowledge and make decisions in real time.

Different and interesting future work is represented by the chance to add more useful information from other sources, such as weather stations, urban sensors, and human wearable devices. Several improvements to the user interface of the current tool provides the chance to upload a custom GTFS data source and a more user-friendly interface that can show the content of the ontology without requiring the user to rely on external viewers, such as Protégé. Additional query languages, different from SPARQL-DL, such as SPARQL or DL query, can be supported.

6. REFERENCES

[1] M. Batty, "Big data, smart cities and city planning," Dialogues Hum. Geogr., vol. 3, no. 3, pp. 274–279, 2013.

[2] S. Allwinkle and P. Cruickshank, "Creating smart-er cities: An overview," J. urban Technol., vol. 18, no. 2, pp. 1–16, 2011.

[3] G. C. Lazaroiu and M. Roscia, "Definition methodology for the smart cities model," Energy, vol. 47, no. 1, pp. 326–332, 2012.

[4] M. Deakin and H. Al Waer, "From intelligent to smart cities," Intell. Build. Int., vol. 3, no. 3, pp. 140–152, 2011.

[5] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," IEEE Internet Things J., vol. 1, no. 1, pp. 22–32, 2014.

[6] R. Kitchin, "Making sense of smart cities: addressing present shortcomings," Cambridge J. Reg. Econ. Soc., vol. 8, no. 1, pp. 131–136, 2015.

[7] C. Manville et al., "Mapping smart cities in the EU," 2014.

[8] G. Hancke, B. Silva, and G. Hancke Jr, "The role of advanced sensing in smart cities," Sensors, vol. 13, no. 1, pp. 393–425, 2013.

[9] M. Abdel-Basset and M. Mohamed, "The role of single valued neutrosophic sets and rough sets in smart city: Imperfect and incomplete information systems," Measurement, vol. 124, pp. 47–55, 2018.

[10] A. Caragliu and C. F. Del Bo, "Smart innovative cities: The impact of Smart City policies on urban innovation," Technol. Forecast. Soc. Change, vol. 142, pp. 373–383, 2019.

[11] D. M. Mejia, Integration of Heterogeneous Traffic Data to Address Mobility Challenges in the City of El Paso. The University of Texas at El Paso, 2017.

[12] A. Caragliu, C. Del Bo, and P. Nijkamp, "Smart cities in Europe," J. urban Technol., vol. 18,

no. 2, pp. 65–82, 2011.

[13] M. Batty et al., “Smart cities of the future,” Eur. Phys. J. Spec. Top., vol. 214, no. 1, pp. 481–518, 2012.

[14] V. Fernandez-Anez, J. M. Fernández-Güell, and R. Giffinger, “Smart City implementation and discourses: An integrated conceptual model. The case of Vienna,” Cities, vol. 78, pp. 4–16, 2018.

[15] L. Anthopoulos, P. Fitsilis, and C. Ziozias, “What is the Source of Smart City Value?: A Business Model Analysis,” in Smart Cities and Smart Spaces: Concepts, Methodologies, Tools, and Applications, IGI Global, 2019, pp. 56–77.

[16] R. Giffinger and H. Gudrun, “Smart cities ranking: an effective instrument for the positioning of the cities?,” ACE Archit. city Environ., vol. 4, no. 12, pp. 7–26, 2010.

[17] M. Angelidou, “Smart cities: A conjuncture of four forces,” Cities, vol. 47, pp. 95–106, 2015.

[18] A. M. Townsend, Smart cities: Big data, civic hackers, and the quest for a new utopia. WW Norton & Company, 2013.

[19] P. Espinoza-Arias, M. Poveda-Villalón, R. García-Castro, and O. Corcho, “Ontological Representation of Smart City Data: From Devices to Cities,” Appl. Sci., vol. 9, no. 1, p. 32, 2019.

[20] V. Albino, U. Berardi, and R. M. Dangelico, “Smart cities: Definitions, dimensions, performance, and initiatives,” J. urban Technol., vol. 22, no. 1, pp. 3–21, 2015.

[21] M. Katsumi and M. Fox, “Ontologies for transportation research: A survey,” Transp. Res. Part C Emerg. Technol., vol. 89, pp. 53–82, 2018.

[22] B. Lorenz, H. J. Ohlbach, and L. Yang, “Ontology of transportation networks,” 2005.

[23] Z. Xiao, H. B. Lim, and L. Ponnambalam, “Participatory sensing for smart cities: A case study on transport trip quality measurement,” IEEE Trans. Ind. Informatics, vol. 13, no. 2, pp. 759–770, 2017.

[24] “GTFS Transit Data,” 2019. [Online]. Available: https://www.sfmta.com/reports/gtfs-transit-data. [Accessed: 19-Apr-2019].

[25] K. M. De Oliveira, F. Bacha, H. Mnasser, and M. Abed, “Transportation ontology definition and application for the content personalization of user interfaces,” Expert Syst. Appl., vol. 40, no. 8, pp. 3145–3159, 2013.

[26] A. J. Bermejo, J. Villadangos, J. J. Astrain, and A. Cordoba, “Ontology based road traffic management,” in Intelligent Distributed Computing VI, Springer, 2013, pp. 103–108.

[27] M. Codescu, G. Horsinka, O. Kutz, T. Mossakowski, and R. Rau, “Osmonto-an ontology of openstreetmap tags,” State map Eur., vol. 2011, 2011.

[28] N. Anand, M. Yang, J. H. R. Van Duin, and L. Tavasszy, “GenCLOn: An ontology for city logistics,” Expert Syst. Appl., vol. 39, no. 15, pp. 11944–11960, 2012.

[29] N. Hill, G. Gibson, E. Guidorzi, S. Amaral, A. Parlikad, and Y. Jin, “Scoping Study into Deriving Transport Benefits from Big Data and the Internet of Things in Smart Cities,” 2017.

[30] P. Rode et al., “Accessibility in cities: transport and urban form,” in Disrupting mobility, Springer, 2017, pp. 239–273.

[31] K. Farkas, G. Feher, A. Benczur, and C. Sidlo, “Crowdsensing based public transport information service in smart cities,” IEEE Commun. Mag., vol. 53, no. 8, pp. 158–165, 2015.

[32] C. Navarro, M. Roca-Riu, S. Furió, and M. Estrada, “Designing new models for energy efficiency in urban freight transport for smart cities and its application to the Spanish case,” Transp. Res. Procedia, vol. 12, pp. 314–324, 2016.

[33] C. F. Calvillo, A. Sánchez-Miralles, and J. Villar, “Energy management and planning in smart cities,” Renew. Sustain. Energy Rev., vol. 55, pp. 273–287, 2016.

[34] C. Lim, K. Kim, and P. P. Maglio, “Smart cities with big data : Reference models , challenges , and considerations,” Cities, no. February, pp. 1–14, 2018.

[35] R. Khatoun and S. Zeadally, “Smart cities: concepts, architectures, research opportunities.,” Commun. Acm, vol. 59, no. 8, pp. 46–57, 2016.

[36] F. Behrendt, “Why cycling matters for smart cities. Internet of bicycles for intelligent transport,” J. Transp. Geogr., vol. 56, pp. 157–164, 2016.

[37] F. Bifulco, M. Tregua, C. C. Amitrano, and A. D’Auria, “ICT and sustainability in smart cities management,” Int. J. Public Sect. Manag., vol. 29, no. 2, pp. 132–147, 2016.

[38] P. Cardullo and R. Kitchin, “Being a ‘citizen’in the smart city: up and down the scaffold of smart citizen participation in Dublin, Ireland,” GeoJournal, vol. 84, no. 1, pp. 1–13, 2019.

[39] C. J. Martin, J. Evans, and A. Karvonen, “Smart and sustainable? Five tensions in the visions and practices of the smart-sustainable city in Europe and North America,” Technol. Forecast. Soc. Change, vol. 133, pp. 269–278, 2018.

[40] “W3C Semantic Web Interest Group: Basic Geo (WGS84 lat/long) Vocabulary.” [Online]. Available: https://www.w3.org/2003/01/geo/. [Accessed: 20-Apr-2019].

[41] S. Roche, “Geographic information science II: Less space, more places in smart cities,” Prog. Hum. Geogr., vol. 40, no. 4, pp. 565–573, 2016.

[42] E. Sirin and B. Parsia, “SPARQL-DL: SPARQL Query for OWL-DL.,” in OWLED, 2007, vol. 258.

[43] “Calculate Distance by Latitude and Longitude using Java | GeoDataSource.” [Online]. Available: http://www.geodatasource.com/developers/java. [Accessed: 19-Apr-2019].