International Journal of Web & Semantic Technology (IJWesT) Vol.15, No.2, April 2024

FFO: FOREST FIRE ONTOLOGY AND REASONING SYSTEM FOR ENHANCED ALERT AND MANAGEMENT SERVICES

Shibani Das and Abhishek Srivastava

Department of Computer Science and Engineering, Indian Institute of Technology Indore, India

ABSTRACT

Forest fires or wildfires pose a serious threat to property, lives, and the environment. Early detection and mitigation of such emergencies, therefore, play an important role in reducing the severity of the impact caused by wildfire. Unfortunately, there is often an improper or delayed mechanism for forest fire detection which leads to destruction and losses. These anomalies in detection can be due to defects in sensors or a lack of proper information interoperability among the sensors deployed in forests. This paper presents a lightweight ontological framework to address these challenges. Interoperability issues are caused due to heterogeneity in technologies used and heterogeneous data created by different sensors. Therefore, through the proposed Forest Fire Detection and Management Ontology (FFO), we introduce a standardized model to share and reuse knowledge and data across different sensors. The proposed ontology is validated using semantic reasoning and query processing. The reasoning and querying processes are performed on real-time data gathered from experiments conducted in a forest and stored as RDF triples based on the design of the ontology. The outcomes of queries and inferences from reasoning demonstrate that FFO is feasible for the early detection of wildfire and facilitates efficient process management subsequent to detection.

KEYWORDS

Semantic Web, Ontology, Semantic Reasoning, Query Processing, SSN,SPARQL, RDF, Geosparql, SWRL, Forest Fire

1. INTRODUCTION

This Forest fires pose a significant threat to human life, property, and the environment. According to the World Wildlife Fund (WWF), increasing fire frequencies and intensities are a hazard to 84% of the surface area of all ecoregions that are essential for maintaining the diversity of species on Earth. The percentage of fires that occur that are within ecologically permissible bounds is only 16% [1]. According to the US National Interagency Fire Center, 56,000 wildfires consumed more than 4.7 million acres of land in 2021 [2]. With over 130,000 fires reported, 2021 saw the highest number of fires in the Amazon rain-forest in ten years. Emergency events such as wildfires can lead to major destruction if they are not detected early or not communicated quickly enough. It is seen that even a slight delay in fire detection can lead to havoc in wildlife and the natural habitat of the forest. It is crucial, therefore, to detect such fires early and to make the right decisions in such emergency situations. Delays in fire detection can be due to several reasons among which difficulty in communication is quite common. Emergency Responders (ERs) use a variety of heterogeneous information obtained from various systems and distinct technologies, which causes communication issues and uncertainty [3]. Over the years, several methods have been developed to detect forest fires, ranging from traditional methods such as human observers

DOI: 10.5121/ijwest.2024.15201

to more advanced technologies like remote sensing and artificial intelligence. But the information about the nearby environment gathered from various observation sources has to be accessed quickly and shared with a central monitoring system. Semantic Web technologies address these interoperability issues with the help of ontologies and information retrieval using SPARQL [4]. Ontologies can facilitate the sharing and reuse of knowledge across different systems and organizations, which can help to improve the overall effectiveness of forest fire detection and mitigation efforts. Ontologies help in making the systems interoperable by standardizing the forest fire-related data used in the different devices, which gather information. The fundamental RDF (Resource Description Framework) triple format (*subject-predicate-object*) is used by ontologies to store information. RDF is the data model for Semantic Web. Through the SPARQL Protocol and RDF Query Language (SPARQL), we are able to query this RDF data and retrieve some valuable observational information including the temporal (time) and the spatial (coordinates) data.

The term "ontology" was defined by T. R. Gruber in 1992 as “explicit specification of a conceptualization” [5]. Ontologies are formal representations of concepts and relationships within a particular domain of knowledge. They give a common understanding of concepts, their meanings, and the connections between them. Ontologies have grown in significance in a variety of disciplines as they allow machines to comprehend the meaning of data and support knowledge sharing and reuse. Ontologies have been the subject of in-depth study in the realm of natural disasters [6]. Many published research focus on developing ontologies that define concepts within specific domains, such as hydrology or wildfire management, as well as generalized domains, along with reasoning rules that allow for the inference of new knowledge from existing data.

Masa et al. [7] describe the ONTO-SAFE framework, which aims to improve the effectiveness of detecting forest fires and offers a decision-support system for managing in the context of wildfire hazards. Based on the SSN vocabulary [8], SoKNOS ontology [9] and also beAWARE ontology [10], the ONTO-SAFE framework uses SHACL-compliant rules [11] as the reasoning scheme. Chandra et al. [12] developed rules using SWRL(Semantic Web Rule Language) [13] for calculating fire weather indices like FFMC, DMC, DC ISI, BUI and FWI [14], which are used to measure fire danger with respect to the prevailing weather conditions. Kalabokidis et al. [6] presented OntoFire, in which they attempted to extract meaningful information in geo-portal environments by employing hyperlinks rather than browsing with keyword-based queries and for that reason, they had to maintain a metadata catalogue.

Wang et al. [15] proposed a hydrological sensor web ontology based on W3C SSN ontology by including the W3C Time Ontology [16] and OGC GeoSPARQL [17]. The BeAWARE[10] ontology is a knowledge representation of concepts relevant to the management of climate-related crises. The ONTOEMERGE (2010-2013) [18], an ontology developed by UFRJ and University of Valencia, contains some generic potential concepts like climatic condition, incident, emergency, organisation, resource, event, among others. The EmergencyFire ontology [19] facilitates standardization and sharing of response protocols for fire in buildings. It facilitates interoperability between people and systems and a reduction in occurrences of false compliances.

Competency Questions (CQs) play a vital role in ontology evaluation. The efficiency of an ontology depends on the answerability of the ontology to the CQs. The QuestionChecker module, which considers CQs expressed as interrogative phrases that work over classes and their relations, is presented by Bezerra et al. along with a discussion of its significance [20]. References [21], [22], [23] have also suggested utilizing CQs while designing ontologies.

Kanellopoulos et al. [30] examine how well several transport layer protocols like UDP, DCCP, SCTP, and TFRC, perform for a range of traffic flows over wired IP networks, including voice over IP (VoIP), video streaming, and data transmission. Wheeb et al. [31] focus on the support for quality of service (QoS) offered by TCP-friendly rate control (TFRC) protocol and Stream Control Transmission Protocol (SCTP) for multimedia streaming applications over Mobile Ad-hoc Networks (MANETs). [32] is primarily concerned with examining Voice over Internet Protocol (VoIP) application performance in wireless networks, with an emphasis on issues related to capacity and QoS. These three articles are of significance to our work due to their respective requirements of QoS.This understanding can help ensure that our system maintains reliable communication and timely response during forest fire incidents.

Table 1. Comparative table to highlight the distinctions between proposed work and existing related work.

Ontology(References)	**Domain**	**Frameworks used**	**Vocabularies or Ontologies used**
BeAWARE [10]	Natural disasters	OntOlogy Pitfall Scanner [35], OntoMetrics.	PESCaDO ontologies [34].
Wang et al. [15]	Hydrological (related to floods)	Protégé.	SSN, GeoSPARQL, Time.
EmergencyFire [19]	Fire in buildings	Protégé.	_
ONTOEMERGE [18]	Climatic conditions	Protégé.	_
OntoFire [6]	Geo-portal about wildfires	Apache Tomcat	_
FFO (proposed ontology)	Forest fire detection (alert) and management	GraphDB, Protégé.	SSN, GeoSPARQL.

In this paper, we present a novel FFO, a lightweight ontology that can be used to interpret sensor data and is intended to standardize the concepts and relationships among the sensors involved in forest fire detection and also provide efficient steps for managing the wildfire. The unique aspect of this work is the ontological architecture specially designed to be applicable to the sensors deployed in Wireless Sensor Networks (WSN) for wildfire monitoring. Concerning the design, we extended the W3C's Semantic Sensor Network (SSN) [8] by introducing new classes for various sensor types required for forest-fire detection, classes to indicate the fire risk levels, as well as by adding new properties to link the classes. Additionally, we instantiated these classes based on the specifications of our experiments conducted in the IIT Indore forest. Our primary research questions include displaying readings in a specific time period, identifying sensor locations, the location of the detected fire, giving information about the population of a particular settlement near the fire location, finding hospitals and fire stations nearby, and more. Overall, the objective is to eradicate the information interoperability issues among the different sensor node technologies, to respond to the detected anomalies in sensors' readings promptly, and to properly manage the wildfire if detected. Table 1 shows the distinctions in the domain of interest between the proposed ontology and the related existing ontologies.'-' in the fourth column of the table means the ontology has not used any other ontologies as base and designed the ontology from scratch. A detailed comparative analysis between the existing work and the proposed work is presented in section 4.

The rest of the paper is organized as follows. Section 2 includes the conceptual model of the proposed ontology with the ontology instantiation and the competency questions (CQs). Section 3 discusses the experimental data, and shows the results and evaluation of the ontology. Section 4 covers the discussion. Section 5 concludes the proposed work and presents the future scope of the work.

2. METHODS

The proposed FFO ontology is designed by extending the standard W3C SSN ontology [8] and OGC GeoSPARQL[17]. The main ontological components involved are the Sensor Ontology (extension of SSN ontology) in which concepts or classes related to sensors and their observations are defined, the Settlement ontology defining the concepts related to settlements, the GeoSPARQL ontology having the concepts related to the location of a point or an area of interest like location of the point where the fire is detected or location of a hospital nearby, etc. i
The proposed ontology was constructed with five main objectives:

1. to define the main concepts, instances and properties (relationships) between the concepts in the Forest Fire Detection and Management domain.
2. to extend the SSN ontology to the forest fire domain and to link the ontologies involved - the sensor ontology, the settlement ontology and GeoSPARQL.
3. to overall monitor efficiently and enable fast response by improving the semantic interoperability among the sensor nodes.
4. to infer new knowledge from the existing data stored and enhance the reasoning process by establishing inference rules and query processing.
5. to perform some emergency management tasks like alerting the authorities of fire stations nearby about the detected fire, finding hospitals nearby, etc.

2.1. The Framework of the Proposed Ontology

Simplicity was the key principle while constructing our ontology and hence, we covered the aspects of the forest fire detection and management system essential to our experiment with the possible minimum number of classes and properties. FFO is designed using the Protégé software [24]. The core classes and properties are shown in figure 1. As we can see in the figure, the properties (predicate) are defined to connect a subject to an object in an RDF triple structure (*subject-predicate-object*). For example, *:HighRish*→ *:Risk*. *:HighRisk*and *:Risk* are both classes. *:HighRisk*is the subjectand *:Risk* is the object.→ denotes the relationship or the property between the subject and the object. Here the property name is*rdfs:subClassOf*.There are two types of properties - 1) Object properties and 2) Data Properties. The object properties are those in which the object is a class whereas data properties are the properties in which the object is a literal or value. In figure 1, we have shown the main object properties. The classes and the properties are discussed thoroughly in sections 2.1.1 and 2.1.2.

The SSN and the GeoSPARQL ontologies are used as the primary base ontologies and also have been extended as well as instantiated.SSN and GeoSPARQL vocabularies were chosen because of their various benefits. The SSN ontology is used to define the sensors, deployment in which the sensors are deployed in the forest, and also the observations and results from the deployed sensors.The SSN ontology can broadly describe sensors, sensing, sensor measurement capabilities, sensing-related observations, and deployments. Due to its wide inclusivity of the concepts related to sensors, the SSN is commonly used as the base ontology for a number of different ontologies designed for a variety of sensors used in different purposes. However, SSN ontology is a domain-independent vocabulary, thereby having the need to extend it in order to design any kind of domain-specific ontology [33]. The OGC GeoSPARQL is used to define vocabulary for expressing geographical data in the ontology as well as establish vocabulary for processing spatial data and reasoning by expanding the SPARQL query language. The base classes from these ontologies have been included and in addition to that, many classes and subclasses have been created according to the domain need in the proposed ontology like *FireStation, Forest, Settlement, CO2level, HumidityValue, SmokeSensor*, etc. as shown in figure 2. The class "*Unit"*has been created in the proposed ontology. Below we discuss the main classes and properties in the proposed ontology. "*prefix:class_name*" and "*prefix:property_name*" notation are used to introduce a class and a property respectively. The prefix denotes the ontology to which the class or the property belongs. No prefix is used for our proposed ontology.

Table 2. Prefixes and namespaces used in the proposed ontology.

Prefix	**Namespace URI**	**Description**
sosa	https://www.w3.org/ns/sosa/	The lightweight Sensor, Observation, Sample, and Actuator (SOSA) ontology, which forms the basis of SSN, aims to broaden the audience for Semantic Web ontologies as well as the scope of applications that can use them.
ssn	https://www.w3.org/ns/ssn/	The Semantic Sensor Network (SSN) ontology is an ontology for describing actuators and sensors, as well as their observations, related processes, interesting topics for research, samples utilized in that research, and observed attributes.
geosparql	http://www.opengis.net/ont/geosparql#	A collection of GeoSPARQL-compatible, domain-specific spatial filter functions for use in SPARQL queries.
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#	An information representation system for the Web is called Resource Description Framework (RDF). In RDF graphs, which are collections of *subject-predicate-object* triples, IRIs, blank nodes, and datatyped literals can all be used as elements. They are used to give descriptions of resources.
rdfs	http://www.w3.org/2000/01/rdf-schema#	For RDF data, RDF Schema offers a vocabulary for data modeling. An expansion of the fundamental RDF vocabulary is RDF Schema.
xsd	http://www.w3.org/2001/XMLSchema#	The structure of an XML document is described by an XML Schema. XML Schema Definition (XSD) is another name for the XML Schema language.
swrlb	http://www.w3.org/2003/11/swrlb	The logic operation formulae for boolean operations, string operations, mathematical computations, etc. are included in built-ins, which are modular SWRL components.

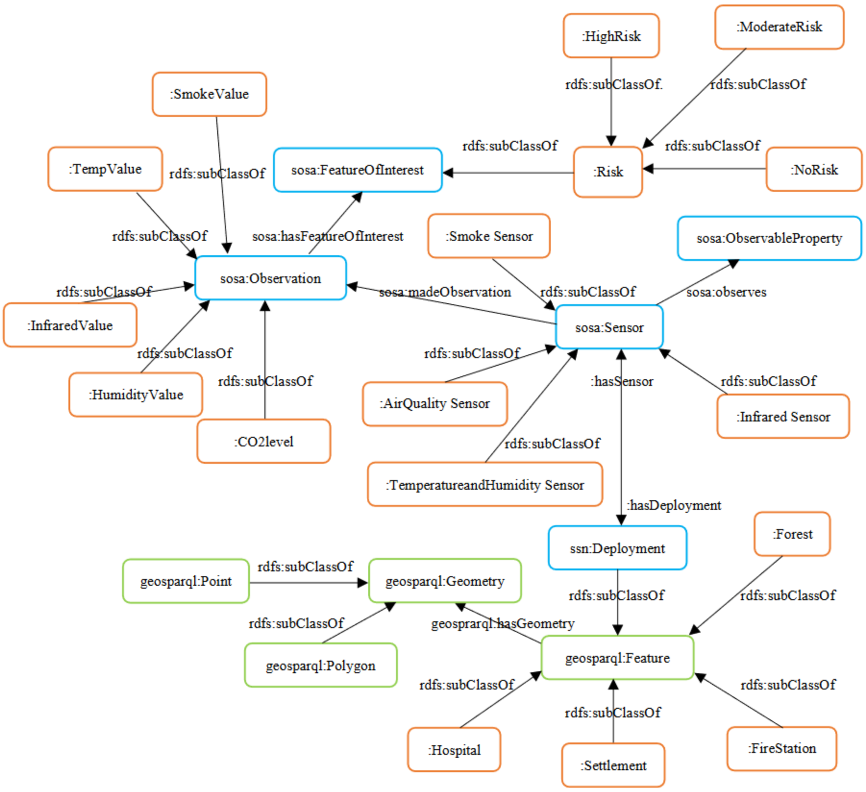


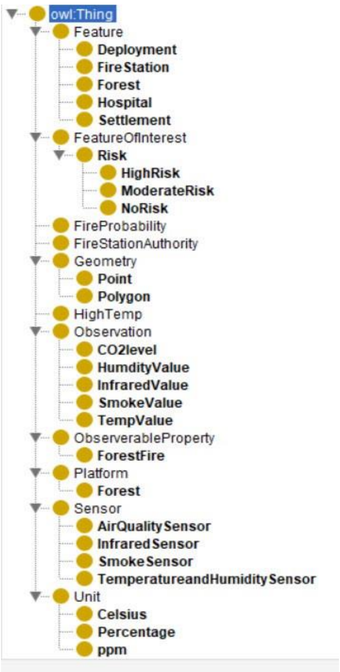
Figure 1.The core classes and properties in FFO based on SSN and GeoSPARQL. The classes from the SSN ontology are outlined with 'blue' colour, those from the GeoSPARQL ontology are outlined with 'green' colour and the classes from our proposed ontology are outlined with 'orange' colour. *prefix:name* notation is also used. No prefix is used for our proposed ontology.

2.1.1. Classes

- **sosa:Sensor**: The class, Sensor, is taken from SOSA ontology (Table 2). It represents all the sensors that we deployed and has four subclasses in our ontology: 1) *TemperatureandHumiditySensor* (to measure temperature and humidity), 2) SmokeSensor (to detect smoke), 3) *AirQualitySensor*(to get the carbon dioxide level), and 4) *InfraredSensor* (to detect movement). DHT11 has been used as temperature and humidity sensor. For smoke sensor, we have used MQ2 gas sensor. MQ135 is used as air quality gas sensor to measure the level of carbon dioxide (CO2) in atmosphere. Finally, IR sensor as an infrared sensor. Altogether, there are 20 sensors (five of each category) under the class *Sensor*. These 20 sensors are the individuals or instances of the class.
- **sosa:Observation**: Class *Observation* has five subclasses in our ontology - *TempValue*, *HumidityValue*, *SmokeValue*, *InfraredValue*, and *CO2level*.
- **geosparql:Feature**: Every entity is a *Feature* if it has a geographical location or area. We have created five subclasses in *Feature*, viz. *Deployment*, *Forest*, *FireStation*, *Hospital*, and *Settlement*.
- **ssn:Deployment**: There are five deployments that have been deployed in the forest of IIT Indore. Each deployment has a set of four sensors DHT11, MQ2, MQ135 and IR sensor. We have created instances (individuals) by the same names for these four sensors in the four subclasses under the class *Sensor* (figure 2). The location of the detected fire can be traced from the location of a deployment whose sensors detect the fire. The location has been recorded by GPS (Global Positioning System) sensor.

- **geosparl:Geometry:** In [25], the OGC GeoSPARQL class *Geometry* is described as a coherent collection of direct positions in space. A spatial reference system (SRS) is used to hold the positions. It has two subclasses: *Point* for one single location of interest, and *Polygon* for an area of interest. These define the coordinates of a location.
-

The class hierarchy is shown in figure 2.

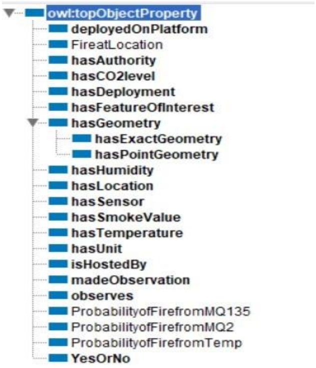


- owl:Thing
  - Feature
    - Deployment
    - FireStation
    - Forest
    - Hospital
    - Settlement
  - FeatureOfInterest
    - Risk
      - HighRisk
      - ModerateRisk
      - NoRisk
  - FireProbability
  - FireStationAuthority
  - Geometry
    - Point
    - Polygon
  - HighTemp
  - Observation
    - CO2level
    - HumdityValue
    - InfraredValue
    - SmokeValue
    - TempValue
  - ObserverableProperty
    - ForestFire
  - Platform
    - Forest
  - Sensor
    - AirQualitySensor
    - InfraredSensor
    - SmokeSensor
    - TemperatureandHumiditySensor
  - Unit
    - Celsius
    - Percentage
    - ppm

Figure 2.Class hierarchy of FFO as shown in Protégé.

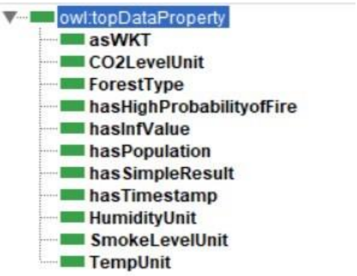
2.1.2. Properties

In this section, the main properties (object properties and data properties) are discussed. As mentioned earlier, properties (predicate) are used to link the subject and the object in the RDF triple (*subject-predicate-object*). *Subject→ Object* notation is used to denote the domain and range of each property. The object properties are shown in figure 3 and the data properties of the proposed ontology are shown in figure 4.



- owl:topObjectProperty
  - deployedOnPlatform
  - FireatLocation
  - hasAuthority
  - hasCO2level
  - hasDeployment
  - hasFeatureOfInterest
  - hasGeometry
    - hasExactGeometry
    - hasPointGeometry
  - hasHumidity
  - hasLocation
  - hasSensor
  - hasSmokeValue
  - hasTemperature
  - hasUnit
  - isHostedBy
  - madeObservation
  - observes
  - ProbabilityofFirefromMQ135
  - ProbabilityofFirefromMQ2
  - ProbabilityofFirefromTemp
  - YesOrNo

Figure 3. Object properties in the proposed ontology (as shown in Protégé).



owl:topDataProperty
- asWKT
- CO2LevelUnit
- ForestType
- hasHighProbabilityofFire
- hasInfValue
- hasPopulation
- hasSimpleResult
- hasTimestamp
- HumidityUnit
- SmokeLevelUnit
- TempUnit

Figure 4.Data properties in the proposed ontology (as shown in Protégé).

- **ssn:deployedOnPlatform**: It is an object property showing the relation between *Deployment* and *Platform*. In our case, *Forest* is a *Platform*. For example, *Deployment-deployedOnPlatform-Forest*.

Deployment → *Platform*

- **:hasCO2Level**: An object property created in the proposed ontology to show the relation between *AirQualitySensor*and *CO2level*.

AirQualitySensor→ *CO2level*

- **:hasDeployment**:showing on which deployment a sensor is placed.

Sensor → *Deployment*

- **:hasSensor**: Inverse property of the property *hasDeployment*. It shows which sensor a deployment has.

Deployment → *Sensor*

- **geosparql:hasGeometry**: The object property, *hasGeometry*from GeoSPARQL defines the spatial representation of a *Feature* (class from GeoSPARQL). It forms the link between GeoSPARQL's*Feature* and *Geometry*, which is basically the coordinates of a location. For example, *Deployment-hasGeometry- LatitudeLongitude*.

Feature → *Geometry*

- **:hasLocation**: We created this object property to directly connect a *Feature* with a location (geosparql: Point) or an area (geosparql:Polygon).

Feature → *Point*
Feature → *Polygon*

- **sosa:madeObservation**: Showing the relation between *Sensor* and *Observation*. This is necessary to display the readings that a sensor provides.

Sensor → *Observation*

- **geosparql:asWKT**: This a data property from GeoSPARQL which links class *Geometry* with the datatype, *wktLiteral*, from GeoSPARQL. The datatype *geosparql:wktLiteral* is used to contain the Well- Known Text (WKT) serialization of a Geometry [25].

Geometry → *wktLiteral*

- **sosa:hasSimpleResult**:This data property links *HumidityValue*or *TempValue*or *CO2level* or *SmokeValue*to *xsd:float*datatype from XSD vocabulary (table 2) to get the output value of each observation from a sensor.

HumidityValue→ *float*
TempValue→ *float*
SmokeValue→ *float*
CO2level → *float*

- **:hasTimestamp**: It links *HumidityValue*or *TempValue*or *CO2level* or *SmokeValue*to xsd:dateTime to get the timestamp of observation.

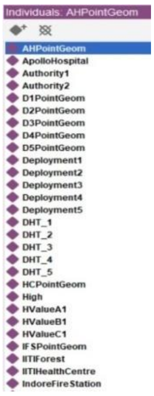
TempValue→ *dateTime*
HumidityValue→ *dateTime*
SmokeValue→ *dateTime*
InfraredValue→ *dateTime*
CO2level → *dateTime*

- **:hasPopulation**: It links class *Settlement* and the datatype*xsd:integer*. It defines the population of a settlement. This is important to know to get an idea of the impact that will be caused by the fire.

Settlement → *integer*

2.1.3. Individuals

Individuals are the concrete entities or instances that exist within a domain and are represented within the ontology. A class may or may not have individual(s). In our ontology, *Deployment1*, *Deployment2*, *Deployment3*, *Deployment4*, and *Deployment5* are the five instances of the class Deployment as we have five systems deployed at five locations in the IIT Indore forest. In each deployment, there are four sensors. For example, *Deployment1* has *DHT_1*, *MQ135_1*, *MQ2_1* and *IR_1* sensors. *DHT_1*, *DHT_2*, *DHT_3*, *DHT_4,* and *DHT_5* as the instances of the DHT sensors. Every deployment has a location that is of class *Point.D1PointGeom* is an individual of the class *Point.D1PointGeom* is the location of *Deployment1*. Some of the individuals are shown in figure 5.



AHPointGeom
ApolloHospital
Authority1
Authority2
D1PointGeom
D2PointGeom
D3PointGeom
D4PointGeom
D5PointGeom
Deployment1
Deployment2
Deployment3
Deployment4
Deployment5
DHT_1
DHT_2
DHT_3
DHT_4
DHT_5
HCPointGeom
High
HValueA1
HValueB1
HValueC1
IFSPointGeom
IITIForest
IITIHealthCentre
IndoreFireStation

Figure 5.An illustrative depiction of select individuals of the proposed ontology.

2.2. Competency Questions

The competency questions (CQs) are the set of natural language questions that an ontology is expected to answer correctly to be efficient and represent ontology needs [20]. Therefore, they play a crucial role in the development of an ontology. To determine the scope of our ontology, we interviewed the domain experts from the Forest Department and the Fire Department of Madhya Pradesh, India. We defined the areas in which the ontology should be answerable, based on the experts' responses from the interviews. The two broad areas of concern were Efficiency in Fire Alert (how much risk is involved) and Management Services (like nearest fire stations and hospitals). Based on the needs of the organisations, we designed eight CQs. Our ontology is evaluated on the basis of these CQs. The CQs are as follows.

1. Show the values of sensors from time t1 to time t2.
2. Find the location of the sensor which recorded values greater than the threshold value.
3. What are the hospitals that are nearby with respect to a location?
4. Which are the nearest fire stations?
5. State whether there is a settlement located near a detected fire location. If yes, what is the distance of the settlement from fire location and what is its population?
6. If there is a probability of fire, state which sensor sensed the probable fire and what is the risk level?
7. Whether there is a high risk of fire, a medium risk, or no risk at all?
8. Notify the fire station authorities about the location where there is high probability of fire.

3. RESULTS AND EVALUATION

A series of semantic querying and reasoning was developed to assess the proposed FFO ontology. The Semantic Web Rule Language (SWRL) [13] was designed to be the standard rule language of the Semantic Web. SWRL helps reasoning with W3C Web Ontology Language (OWL) individuals by allowing the users to write rules expressed in terms of OWL concepts. We devised 14 SWRL rules using Protégé (version 5.6.1), and for semantic querying, we used the Semantic Web query language, SPARQL [4] which we have implemented in GraphDB[26]. The SWRL rules for reasoning and queries for semantic query-result processes were designed on the basis of competency questions discussed in section 2.2. The reasoner deduces new inferences based on the existing OWL knowledge bases [27] and the rules defined. The query-answering and reasoning processes will be discussed in this section.

3.1. Experimental Data

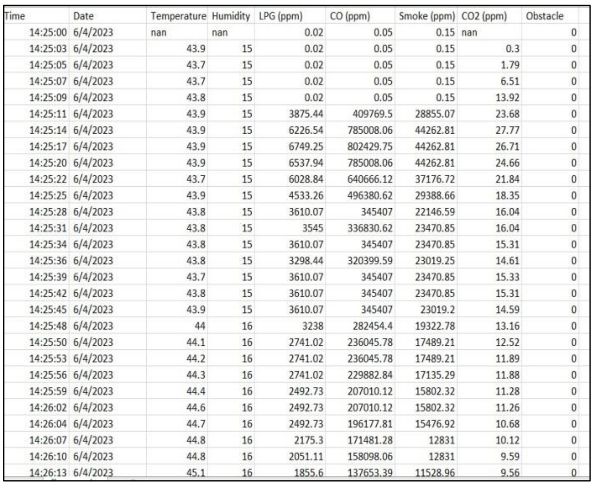
We have collected real-time data by experimenting in the IIT Indore forest. We placed one deployment in each of the five locations and initiated a controlled fire in the forest area near Deployment 3 and collected the readings for different sensors in the deployment. Figure 6shows the recorded readings of deployment 3 and figure 7 shows the placement of deployment 3 in the forest and the initiated fire.

MQ2 smoke sensor exhibited 0.15 ppm (parts per million) when no smoke is detected and a value range from 1200 ppm to 400000 ppm when smoke is detected. MQ135 (CO2 level) gas sensor recorded values in a range of 0.3 - 138 ppm. However, the DHT11 temperature and humidity sensor showed a gradual increase in temperature and humidity. The temperature range was recorded to be from 43.1 °C to 59.2 °C in the presence of fire whereas the normal temperature at the time of the experiment was 38 °C.

We have defined certain thresholds for the DHT11 temperature value, MQ2 smoke level, and MQ135 ppm value for the "Fire Alert" based on the findings of our experiment in the IIT Indore forest. The thresholds will be discussed in the subsection 3.3.

3.2. Query Processing

Queries are processed in GraphDB. SPARQL is the querying language used to query and validate ontologies and it was implemented using Protégé SPARQL Plugin in Protégé. We test the functionality of an ontology using query processing to check its capability to answer as expected to the competency questions.



Time	Date	Temperature	Humidity	LPG (ppm)	CO (ppm)	Smoke (ppm)	CO2 (ppm)	Obstacle
14:25:00	6/4/2023	nan	nan	0.02	0.05	0.15	nan	0
14:25:03	6/4/2023	43.9	15	0.02	0.05	0.15	0.3	0
14:25:05	6/4/2023	43.7	15	0.02	0.05	0.15	1.79	0
14:25:07	6/4/2023	43.7	15	0.02	0.05	0.15	6.51	0
14:25:09	6/4/2023	43.8	15	0.02	0.05	0.15	13.92	0
14:25:11	6/4/2023	43.9	15	3875.44	409769.5	28855.07	23.68	0
14:25:14	6/4/2023	43.9	15	6226.54	785008.06	44262.81	27.77	0
14:25:17	6/4/2023	43.9	15	6749.25	802429.75	44262.81	26.71	0
14:25:20	6/4/2023	43.9	15	6537.94	785008.06	44262.81	24.66	0
14:25:22	6/4/2023	43.7	15	6028.84	640666.12	37176.72	21.84	0
14:25:25	6/4/2023	43.9	15	4533.26	496380.62	29388.66	18.35	0
14:25:28	6/4/2023	43.8	15	3610.07	345407	22146.59	16.04	0
14:25:31	6/4/2023	43.8	15	3545	336830.62	23470.85	16.04	0
14:25:34	6/4/2023	43.8	15	3610.07	345407	23470.85	15.31	0
14:25:36	6/4/2023	43.8	15	3298.44	320399.59	23019.25	14.61	0
14:25:39	6/4/2023	43.7	15	3610.07	345407	23470.85	15.33	0
14:25:42	6/4/2023	43.8	15	3610.07	345407	23470.85	15.31	0
14:25:45	6/4/2023	43.9	15	3610.07	345407	23019.2	14.59	0
14:25:48	6/4/2023	44	16	3238	282454.4	19322.78	13.16	0
14:25:50	6/4/2023	44.1	16	2741.02	236045.78	17489.21	12.52	0
14:25:53	6/4/2023	44.2	16	2741.02	236045.78	17489.21	11.89	0
14:25:56	6/4/2023	44.3	16	2741.02	229882.84	17135.29	11.88	0
14:25:59	6/4/2023	44.4	16	2492.73	207010.12	15802.32	11.28	0
14:26:02	6/4/2023	44.6	16	2492.73	207010.12	15802.32	11.26	0
14:26:04	6/4/2023	44.7	16	2492.73	196177.81	15476.92	10.68	0
14:26:07	6/4/2023	44.8	16	2175.3	171481.28	12831	10.12	0
14:26:10	6/4/2023	44.8	16	2051.11	158098.06	12831	9.59	0
14:26:13	6/4/2023	45.1	16	1855.6	137653.39	11528.96	9.56	0

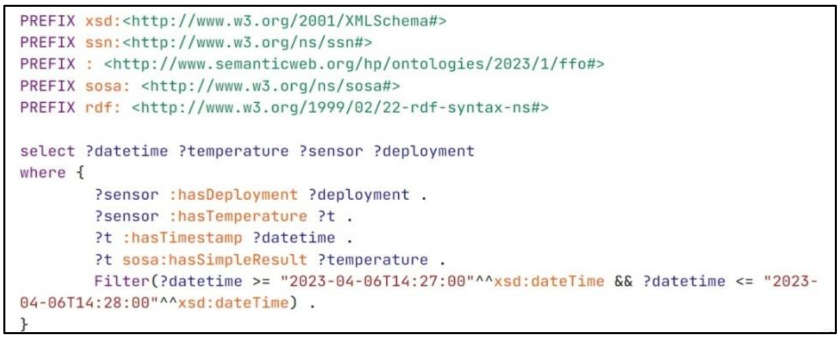
Figure 6. The date and time and the values from the experiment recorded by Deployment 3. The sudden increase in the values implies detected fire.

Figure 7. Deployment3 in the forest of IIT Indore, taking readings in the presence of fire.

- **Query 1**: Query to show the temperature values of DHT sensors from all the deployments within the specified time period. This query corresponds to competency question1 discussed in section 2.2. Figures8 and 9 shows the query and the results respectively.

- **Query 2**: Query(table 3) to display all the hospitals that are within the range of 20 km from the MQ2 sensor which detected smoke level greater than 1200 ppm. We have given this threshold based on the readings recorded in the experiment. This is related to competency question 3. Figure 10 shows the result of query 2.

By processing queries in this manner, we were also able to test the ontology for the aforementioned competency questions 2, 4, and 5 discussed in section 2.2. The results were as we had anticipated.



```
PREFIX xsd:<http://www.w3.org/2001/XMLSchema#>
PREFIX ssn:<http://www.w3.org/ns/ssn#>
PREFIX : <http://www.semanticweb.org/hp/ontologies/2023/1/ffo#>
PREFIX sosa: <http://www.w3.org/ns/sosa#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

select ?datetime ?temperature ?sensor ?deployment
where {
        ?sensor :hasDeployment ?deployment .
        ?sensor :hasTemperature ?t .
        ?t :hasTimestamp ?datetime .
        ?t sosa:hasSimpleResult ?temperature .
        Filter(?datetime >= "2023-04-06T14:27:00"^^xsd:dateTime && ?datetime <= "2023-
04-06T14:28:00"^^xsd:dateTime) .
}
```

Figure 8. Frame view of GraphDB showing the SPARQL query to display the temperature values in DHT sensors of all the deployments in the specified time period (Query 1).

	datetime	temperature	sensor	deployment
1	"2023-04-06T14:27:00"^^xsd:dateTime	"41.8"^^xsd:float	:DHT_2	:Deployment2
2	"2023-04-06T14:27:02"^^xsd:dateTime	"40.0"^^xsd:float	:DHT_1	:Deployment1
3	"2023-04-06T14:27:03"^^xsd:dateTime	"39.0"^^xsd:float	:DHT_2	:Deployment2
4	"2023-04-06T14:27:03"^^xsd:dateTime	"47.6"^^xsd:float	:DHT_3	:Deployment3
5	"2023-04-06T14:27:09"^^xsd:dateTime	"39.6"^^xsd:float	:DHT_4	:Deployment4

Figure 9. Frame view of GraphDBshowing initial five results to the query.

Table 3. Query 2.

```
PREFIX xsd:<http://www.w3.org/2001/XMLSchema#>
PREFIX geof:<http://www.opengis.net/def/function/geosparql/>
PREFIX ssn:<http://www.w3.org/ns/ssn#>
PREFIX : <http://www.semanticweb.org/hp/ontologies/2023/1/final#>
PREFIX geosparql: <http://www.opengis.net/ont/geosparql#>
PREFIX sosa: <http://www.w3.org/ns/sosa#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
select ?deployment ?sensor ?smoke_value ?Hospital ?distance where {
?deployment a ssn:Deployment .
?deployment :hasSensor ?sensor .
?sensor :hasSmokeValue ?tvalue.
?tvaluesosa:hasSimpleResult ?smoke_value .
:Deployment3 :hasLocation ?d1 .
?d1 geosparql:asWKT ?l1 .
?Hospital a :Hospital .
?Hospital :hasLocation ?h .
?h geosparql:asWKT ?l2 .
BIND(geof:distance(?l1,?l2) as ?distance) .
FILTER(?smoke_value>1200 && ?distance <= 20000).
}
```

	deployment	sensor	smoke_value	Hospital	distance
1	:Deployment3	:MQ2_3	"2371.4"^^xsd:float	:IITIHealthCentre	"432.5770665987057"^^xsd:double
2	:Deployment3	:MQ2_3	"2371.4"^^xsd:float	:MhowRailwayHospital	"16568.84776703363"^^xsd:double

Figure 10. Frame view of GraphDB producing the result to query 2 (discussed in subsection 3.2)

3.3. Rule-based Reasoning

Based on the competency questions mentioned above, we set up rules with the purpose of making inferences and acquiring new knowledge based on the existing classes and relationships (OWL knowledge base), to check whether our ontology can answer the competency questions. This is also a process of evaluating the ontology in addition to query processing.

We used the SWRL language [13] for designing the rules. The rules were implemented by SWRLTab, a plugin for rule specification in Protégé. For reasoning, we used the Pellet Reasoner [28]. We have created 14 Rules for reasoning. The main rules are shown as follows.

1. *swrlb:greaterThanOrEqual (?t, 45) ∧hasDeployment(?s, ?d) ∧hasLocation(?d, ?p) ∧geosparql:asWKT(?p, ?loc) ∧hasTemperature(?s, ?temp) ∧sosa:hasSimpleResult(?temp, ?t)→ ProbabilityofFirefromTemp(?p, High)*
2. *hasSomkeValue(?s, ?sv) ∧hasDeployment(?s, ?d)∧sosa: hasSimpleResult(?sv, ?ss)∧hasLocation(?d, ?p) ∧geosparql: asWKT (?p, ?loc) ∧swrlb:greaterThanOrEqual(?ss, 30000) → ProbabilityofFirefromMQ2(?p, High)*
3. *swrlb: greaterThanOrEqual(?v, 20) ∧hasDeployment(?s, ?d) ∧sosa: hasSimpleResult(?c, ?v) ∧hasLocation(?d, ?p) ∧geosparql: asWKT(?p, ?loc)∧hasCO2level(?s, ?c) → ProbabilityofFirefromMQ135(?p, High)*
4. *ProbabilityofFirefromTemp(?p, High) ∧ProbabilityofFirefromMQ2(?p, High) ∧ProbabilityofFirefrmoMQ135(?p, High) → HighRisk(?p)*
5. *HighRisk(?p)∧hasAuthority(?f, ?a) → FireatLocation(?a, ?p)*

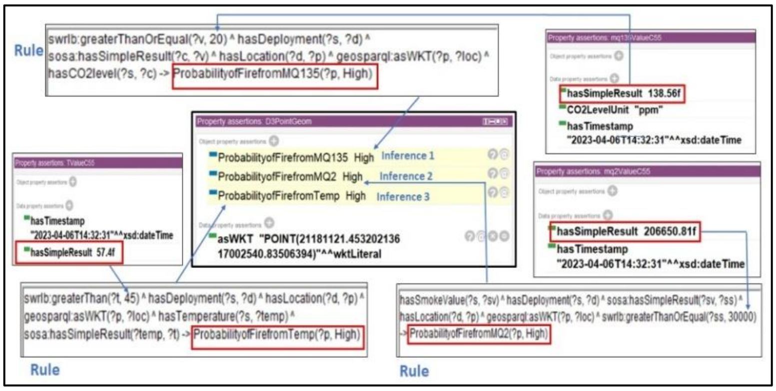
We have defined the threshold values in the rules for the reasoner to draw inferences from the sensors' readings. These threshold values are defined by analysing the nature of readings (figure6) recorded by the sensors deployed in the presence of fire. We have used the Pellet reasoner to deduce inferences from our existing data, and our designed SWRL rules. There were three distinct inferences in our ontology when the reasoner was started. These are discussed below.

- **Reasoning 1**: DHT_3 of Deployment3 recorded 57.4 °C which is greater than 45 °C. According to Rule 1 of the rules discussed above, if the temperature is greater than 45 °C, then the object property called *ProbabilityofFirefromTemp*connects the "location of the Deployment which has the DHT11 sensor" to*High*. In our case, it connects *D3PointGeom*, which is the location of Deployment 3 to the Risk level, *High*. This is the inference 3 in figure 11. Similarly, inferences 1 and 2 are drawn from Rules 3 and 2 mentioned above.
- **Reasoning 2**: If the probability of Fire from DHT11, MQ2, and MQ135 sensors are high, then the location of the corresponding deployment is marked as*HighRisk*. Rule 4 states this deduction. Figure 12shows how the reasoner marks *D3PointGeom* as *HighRisk*as it deduced the inferences discussed in the Reasoning 1 section.
- **Reasoning 3**: If a location is marked as *HighRisk*, then the authorities of all the fire stations should be notified about that location. Rule 5 is designed for this deduction. Figure 13

shows the two fire stations in our ontology which are *IndoreFireStation*and *MhowFireStation*, and *Authority1* and *Authority2* are their authorities respectively. The object property*FireatLocation*links the classes *FireStationAuthority* and *Geometry* tonotify the fire station authorities about the fire location.

Authority1 → D3PointGeom
Authority2 → D3PointGeom

These are the inferences 1 and 2 respectively in Figure 13.



```
swrlb:greaterThanOrEqual(?v, 20) ^ hasDeployment(?s, ?d) ^
sosa:hasSimpleResult(?c, ?v) ^ hasLocation(?d, ?p) ^ geosparql:asWKT(?p, ?loc) ^
hasCO2level(?s, ?c) -> ProbabilityofFirefromMQ135(?p, High)
```

hasSimpleResult 138.56f
CO2LevelUnit "ppm"
hasTimestamp "2023-04-06T14:32:31"^^xsd:dateTime

ProbabilityofFirefromMQ135 High
ProbabilityofFirefromMQ2 High
ProbabilityofFirefromTemp High
asWKT "POINT(21181121.453202136 17002540.83506394)"^^wktLiteral

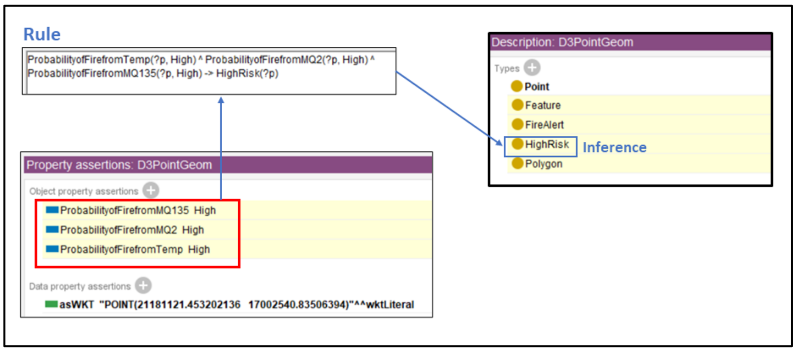
hasTimestamp "2023-04-06T14:32:31"^^xsd:dateTime
hasSimpleResult 57.4f

hasSimpleResult 206650.81f
hasTimestamp "2023-04-06T14:32:31"^^xsd:dateTime

```
swrlb:greaterThan(?t, 45) ^ hasDeployment(?s, ?d) ^ hasLocation(?d, ?p) ^
geosparql:asWKT(?p, ?loc) ^ hasTemperature(?s, ?temp) ^
sosa:hasSimpleResult(?temp, ?t) -> ProbabilityofFirefromTemp(?p, High)
```

```
hasSmokeValue(?s, ?sv) ^ hasDeployment(?s, ?d) ^ sosa:hasSimpleResult(?sv, ?ss) ^
hasLocation(?d, ?p) ^ geosparql:asWKT(?p, ?loc) ^ swrlb:greaterThanOrEqual(?ss, 30000)
-> ProbabilityofFirefromMQ2(?p, High)
```

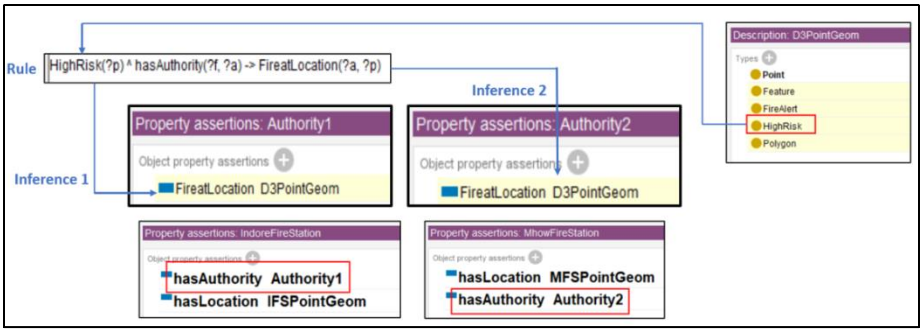
Figure 11. Protégé frame view showing inferences deduced by the reasoner with respect to the experiment data values of Deployment 3 and the rules 1,2,3 given in section 3.3.



```
ProbabilityofFirefromTemp(?p, High) ^ ProbabilityofFirefromMQ2(?p, High) ^
ProbabilityofFirefromMQ135(?p, High) -> HighRisk(?p)
```

ProbabilityofFirefromMQ135 High
ProbabilityofFirefromMQ2 High
ProbabilityofFirefromTemp High
asWKT "POINT(21181121.453202136 17002540.83506394)"^^wktLiteral

Figure 12. Protégé frame view showing Reasoning 2 which infers that*D3PointGeom* is at *HighRisk*,



```
HighRisk(?p) ^ hasAuthority(?f, ?a) -> FireatLocation(?a, ?p)
```

FireatLocation D3PointGeom
FireatLocation D3PointGeom
hasAuthority Authority1
hasLocation IFSPointGeom
hasLocation MFSPointGeom
hasAuthority Authority2

Figure 13. Protégé frame view showing the authorities of the corresponding fire stations and these authorities are being notified about *D3PointGeom*, the location of Deployment 3 through inferences 1 and 2.

In summary, we evaluated our ontology through query processing and semantic reasoning. The query processing shows that we have successfully retrieved the temperature values of DHT sensors from all deployments within a specific time period. This helps in early detection of sudden rise in the temperature values (if any). The display of all the nearby hospitals from a specific sensor helps in the management process subsequent to the fire detection, which is one of the major objectives of our proposed ontology. The semantic reasoning also helps significantly if there is a sudden rise in the sensor readings. The reasoning process automatically generates inferences from the readings based on the rules we have designed, which can help sending an alert without even human interference if necessary.

4. DISCUSSION

The results of the semantic query processing and the rule-based reasoning were as we had anticipated, which is discussed in the section 3. The ontological architecture of this work is unique since it was designed specifically to be used with the heterogeneous sensors used in Wireless Sensor Networks (WSN) for monitoring wildfires. The proposed ontology was able to minimise the interoperability issues in information exchange among the sensors to a great extent, which helped to monitor the wildfire efficiently and dynamically. We also had initiated a controlled fire in the forest under the supervision of experts and was able to record the data and with that information being communicated dynamically, along with the help of proper semantic query processing and semantic reasoning, we could find the nearest fire stations and hospitals, and also notify the authorities. We also have the *Settlement* class in our ontology which maintains the population record staying nearby.

BeAWARE ontology [10] (refer to table 1), whichis designed for natural disasters, and ONTOEMERGE by UFRJ and University of Valencia [18], whichis designed for climatic conditions, are too generic and contain generalized concepts related to natural disasters and weather respectively. EmergencyFire[19] ontology is designed to have concepts related to fire in buildings, which is a specific domain. Wang et al. [15] designed a hydrological ontology to predict floods. OntoFire [6] is a geo-portal for wildfires designed using ontological approach. It helps its users to find resources in a geographical area or region during natural disasters like wildfire. The proposed ontology, FFO is more lightweight and specific to forest fire detection and management having the most inclusive, SSN ontology and GeoSPARQL as the base ontologies. Furthermore, the compatibility of FFO with Wireless Sensor Networks (WSN) to detect wildfire proves its importance and also justifies its benefits over the existing methods. From the perspective of architecture, many of the existing work are designed from scratch and no other ontologies are taken as base ontologies. Ontologies are based on knowledge sharing and reusing. Therefore, it is a good practice to reuse an ontology and extending it further for a specific domain. As it is shown in table 1, [10] and [15] have reused pre-existing ontologies. FFO also extended SSN and GeoSPARQL vocabularies to be designed specifically for the forest fire detection and management purposes. For the purpose of management, FFO incorporates logical rules to notify nearby fire stations in case of fire detection, as well as the ability to locate hospitals within a specified distance through semantic querying, among other management features. This ensures that our ontological model not only detects fires but also facilitates efficient management afterward. Furthermore, the inclusion of GeoSPARQL ontology along with the SSN ontology facilitates the representation of a place or area of interest in geospatial data in addition to interpret sensor data among the sensor nodes deployed in the forest.

Semantic query processing allows the acquisition of any sensor's readings at any moment or for any duration as shown in section 3.2. Also, we can see the nearest hospitals by processing the appropriate query language. Based on the risk degree of forest fire, we can use semantic reasoning to categorize a site as either High Risk, Moderate Risk, or No Risk Probability. This reasoning takes place dynamically as the deployed sensors take readings from the environment, which marks its novelty as an ontology to be used in Wireless Sensor Networks for efficient monitoring of wildfire. Overall, the proposed ontology, FFO serves as an efficient ontology for monitoring forest fire and a reasoning system for fire alert and management services subsequent to the fire detection. FFO is a forest fire monitoring and management specific extension to W3C SSN ontology. This ontology can be further extended to other forest fire related domains according to the demands of the research.

5. CONCLUSION

In this paper, we have presented the ontology-based model for Forest Fire Detection and Management with the purpose of (a) representing the main concepts and properties of the Forest Fire domain and also instantiating the concepts designed according to the requirements of our experiments, (b) standardizing the data created by sensors deployed in forest and enhancing efficiency in data sharing and reusing among the sensors, thereby improving information interoperability to a great extent, (c) detecting the Risk Level or Fire Probability at the proper time, and, (d) taking actions for systematic management subsequent to fire detection. This paper also discusses how our ontology satisfies the practical feasibility test against the competency questions using semantic reasoning and query processing on the real-time data collected through experiments and stored in RDF triple format according to the design of FFO.

This research work has multiple potential avenues for future research. This is a lightweight ontological prototype in the wildfire monitoring and management domain and can be extended further based on the requirements as future scope. More rules can be added for extensive reasoning and therefore, management steps to be increased further. Further research endeavours may concentrate on broadening the ontology to incorporate additional data sources, such satellite imagery, weather forecasts, and social media feeds, in order to enhance the comprehension of wildfire dynamics and enable more all-encompassing oversight and decision-making procedures. Furthermore, the ontology proposed in this paper, will be uploaded to professional ontology-sharing websites such as the Linked Open Vocabularies for Internet of Things (LOV4IoT) [29].

ACKNOWLEDGEMENTS

This work is supported in part by the Ministry of Education, Government of India and the DeFries-Bajpai Foundation, USA

REFERENCES

[1] Forests Ablaze. Available: https://www.wwf.de/fileadmin/fm-wwf/Publikationen-PDF/WWF-Study-Forests-Ablaze.pdf. (accessed: 30 December 2023).

[2] Wildfire and Acres. Available online: https://www.nifc.gov/fire-information/statistics/wildfires. (Accessed: 30 December, 2023).

[3] V. Nunavath, A. Prinz, Data sources handling for emergency management: Supporting information availability and accessibility for emergency responders, 2017, pp. 240-259. doi:10.1007/978-3-319-58524-6_21.

[4] A. Seaborne, E. Prud'hommeaux, Sparql query language for rdf, W3c recommendation, World Wide Web Consortium (W3C) (2013).URL https://www.w3.org/tR/rdf-sparql-query/

[5] T. R. Gruber, Toward principles for the design of ontologies used for knowledge sharing?, Int. J. Hum. Comput. Stud. 43 (1995) 907-928.

[6] K. Kalabokidis, A. Nikos, M. Vaitis, Ontofire: An ontology-based geo-portal for wildfires, Natural Hazards and Earth System Sciences 11 (2011) 3157-3170. doi:10.5194/nhess-11-3157-2011

[7] P. Masa, G. Meditskos, S. Kintzios, S. Vrochidis, I. Kompatsiaris, Ontology-based modelling and reasoning for forest fire emergencies in resilient societies, in: Proceedings of the 12th Hellenic Conference on Artificial Intelligence, SETN ’22, Association for Computing Machinery, New York, NY, USA, 2022. doi:10.1145/3549737.3549765.URL https://doi.org/10.1145/3549737.3549765

[8] M. Compton, P. Barnaghi, L. Bermudez, R. Garcia-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W. D. Kelsey, D. Le Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, K. Page, A. Passant, A. Sheth, K. Taylor, The ssn ontology of the w3c semantic sensor network incubator group, Journal of Web Semantics 17 (2012) 25-32. doi:https://doi.org/10.1016/j.websem.2012.05.003. URL https://www.sciencedirect.com/science/article/pii/s1570826812000571

[9] G. Babitski, S. Bergweiler, O. Grebner, D. Oberle, H. Paulheim, F. Probst, Soknos – using semantic technologies in disaster management software, in: G. Antoniou, M. Grebelnik, E. Simperl, B. Parsia, D. Plexousakis, P. De Leenheer, J. Pan (Eds.), The Semantic Web: Research and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 183-197.

[10] E. Kontopoulos, P. Mitzias, J. Mobgrader, P. Hertweck, H. van der Schaaf, D. Hilbring, F. Lombardo, D. Norbiato, M. Ferri, A. Karakostas, S. Vrochidis, I. Kompatsiaris, Ontology-based representation of crisis management procedures for climate events, 2018.

[11] H. Knublauch, D. Kontokostas, Shapes constraint language (shacl), W3.org https://www.w3.org/TR/shacl/, [Accessed: March 21, 2023] (2017).

[12] R. Chandra, K. Abhishek, S. Agarwal, N. Singh, Semantic sensor network ontology based decision support system for forest fire management, Ecological Informatics 72 (2022) 101821. doi:https://doi.org/10.1016/j.ecoinf.2022.101821 URL https://www.sciencedirect.com/science/articles/pii/s1574954122002710

[13] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, Swrl: A semantic web rule language, W3.org https://www. w3.org/Submission/SWRL/, [Accessed: March 21, 2023] (2004).

[14] National Wildfire Coordinating Group, Fire weather index system, https://www.nwcg.gov/publications/pms437/cffdrs/ fire- weather- index- system, accessed: March 22, 2023 (2016).

[15] Wang, N. Chen, W. Wang, Z. Chen, A hydrological sensor web ontology based on the ssn ontology: A case study for a flood, ISPRS International Journal of Geo-Information 7 (1) (2018). doi:10.3390/ ijgi7010002.URL https://www.mdpi.com/2220-9964/7/1/2

[16] World Wide Web Consortium. Time Ontology in OWL. Available online: https://www.w3 454.org/TR/owl-time/, 2017. (accessed on December 2023).

[17] M. Perry, J. Herring, Geosparql – a geographic query language for rdf data, OGC.org https://www.ogc.org/standard/geosparql/, (Accessed: March 21, 2023).

[18] M. Bosca, K. Cordeiro, J. Oliveira, M. Campos, Ontoemerge: Construção de umaontologia core para a área de emergênciasbaseademontologia de fundamentação (2010).

[19] K. BITENCOURT, F. Durão, M. Mendonça, L. Santana, An ontological model for fire emergency situations (01 2018). doi:10.1587/ transinf.2017SWP0003.

[20] C. Bezerra, F. Freitas, F. Santana, Evaluating ontologies with competency questions, in: 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), Vol. 3, 2013, pp. 284-285. doi:10.1109/WI-IAT.2013.199.

[21] M. Uschold, M. Gruninger, Ontologies: principles, methods and applications, The Knowledge Engineering Review 11 (2) (1996) 93–136. doi:10.1017/S026988890000779.

[22] M. Fernández-López, A. Gómez-Pérez, N. Juristo, Methontology: From ontological art towards ontological engineering, in: AAAI Conference on Artificial Intelligence, 1997.

[23] P. Haase, H. Lewen, R. Studer, D. Tran, M. Erdmann, M. d’Aquin, E. Motta, The neon ontology engineering toolkit, in: The Web Conference, 2008.

[24] Stanford Center for Biomedical Informatics Research, Protege, https: //protege.stanford.edu/(accessed 2023).

[25] Open Geospatial Consortium, GeoSPARQL - A Geographic Query Language for RDF Data, https://opengeospatial.github.io/ogc- geosparql/geosparql11/index.html, accessed: March 30, 2023 (2012).

[26] Ontotext AD, GraphDB: A semantic graph database, https:// graphdb.ontotext.com/, [Online; accessed 22-March-2023] (2021).

[27] O’Connor, M.J.; Knublauch, H.; Tu, S.W.; Musen, M.A. Writing Rules for the Semantic Web Using SWRL and Jess. 2005.

[28] W3C, Pellet - semantic web standards, https://www.w3.org/2001/sw/ wiki/Pellet, accessed: April 3, 2023

[29] LOV4IoT Project. Available online: http://lov4iot.appspot.com/. (accessed on 24 December2023).

[30] Kanellopoulos, Dimitris &Wheeb, Ali. (2020). Simulated Performance of TFRC, DCCP, SCTP, and UDP Protocols Over Wired Networks. International Journal of Interdisciplinary Telecommunications and Networking. 12. 88-103. 10.4018/IJITN.2020100107.

[31] Wheeb, Ali &Kanellopoulos, Dimitris. (2020). Simulated Performance of SCTP and TFRC Over MANETs: The Impact of Traffic Load and Nodes Mobility. International Journal of Business Data Communications and Networking. 16. 10.4018/IJBDCN.2020070104.

[32] Wheeb, Ali. (2017). Performance Analysis of VoIP in Wireless Networks. International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN: 2250-3501. 7. 1-5.

[33] Calbimonte, Jean-Paul &Jeung, Hoyoung&Corcho, Oscar &Aberer, Karl. (2012). Enabling Query Technologies for the Semantic Sensor Web. International Journal On Semantic Web and Information Systems (IJSWIS). 8. 43-63. 10.4018/jswis.2012010103.

[34] Rospocher, M., Serafini, L. (2013). An Ontological Framework for Decision Support. In: Takeda, H., Qu, Y., Mizoguchi, R., Kitamura, Y. (eds) Semantic Technology. JIST 2012. Lecture Notes in Computer Science, vol 7774. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-37996-3_16

[35] Poveda-Villalón, María & Gomez-Perez, Asuncion & Suárez-Figueroa, Mari Carmen. (2014). OOPS! (OntOlogy Pitfall Scanner!): An on-line tool for ontology evaluation. International Journal on Semantic Web and Information Systems. 10. 7-34. 10.4018/ijswis.2014040102.

AUTHORS

Shibani Das is currently pursuing MS(Research) in the Discipline of Computer Science and Engineering at the Indian Institute of Technology Indore, India. She completed her B.Tech in 2020 from Tezpur University, India. Her main research interests include Semantic Web, Ontologies, Knowledge Graphs, Knowledge Representation, and Natural Language Processing.

Abhishek Srivastava is a Professor in the Discipline of Computer Science and Engineering at the Indian Institute of Technology Indore. He completed his Ph.D. in 2011 from the University of Alberta, Canada. Abhishek's group at IIT Indore has been involved in research on service-oriented systems most commonly realized through web services. More recently, the group has been interested in applying these ideas in the realm of the Internet of Things. The ideas explored include coming up with technology-agnostic solutions for seamlessly linking heterogeneous IoT deployments across domains. Further, the group is also delving into utilizing Machine Learning adapted for constrained environments to effectively make sense of the huge amounts of data that emanate from the vast network of IoT deployments.