xR4DRAMA

Extended Reality For DisasteR management And Media planning

H2020-952133

D3.11

Semantic representation, fusion and reasoning-based decision support system for situation awareness

Dissemination level:	Public
Contractual date of delivery:	Month 27, 31 January 2023
Actual date of delivery:	Month 27, 31 January 2023
Work package:	WP3 - Analysis and fusion of multi-modal data
Task:	T3.5 - Multimodal information fusion T3.7 - Decision support system for media production and disaster management
Type:	Demonstrator
Approval Status:	Final version
Version:	1.0
Number of pages:	51
Filename:	D3.11_xR4Drama_SemanticRepresentationFusionDSS_20230131_v1.0.pdf

Abstract

Based on the requirements structured by WP6 and the dependencies incurring from the interaction with the other WPs (Analysis and fusion of multi-modal data, Platform development, Point Management Mechanism, Decision Support System), the purpose, scope,

intended users and uses, and the requirements of the xR4DRAMA ontology were identified. These specifications, along with the modelling understanding from relevant study fields, played an important guidance role for building the first version of the xR4DRAMA ontology that currently comprises modules for capturing the analysis results from the other modules of WP3 (visual analysis, stress level detection and textual analysis). Furthermore, it describes the population process of these incoming data to the repository of the ontology and presents some validation examples. Next, information retrieval mechanisms that are used for text generation and general information displacement, such as emergency numbers, general location description, and related general documents, are developed for the requirements of WP3. A Point of interest management mechanism which creates or updates points of interest, based on the visual and textual analysis content was also developed for the requirements of T3.5. Finally, the integration of a decision support system that assigns task based on events and asserts severity scores on danger zones, was also developed to provide more compact information to first responders and journalists.

co-funded by the European Union

History

Version	Date	Reason	Revised by
0.1	07/12/22	First release of the template. ToC and assignments finalisation	Pistola Theodora
0.5	20/01/23	Draft for internal review by UPF	Pistola Theodora
1.0	31/01/23	Final version	Pistola Theodora

Author list

Organization	Name	Contact Information
CERTH	Spyridon Symeonidis	spyridons@iti.gr
CERTH	Sotiris Diplaris	diplaris@iti.gr
CERTH	Georgios Tzanetis	tzangeor@iti.gr
CERTH	Pistola Theodora	tpistola@iti.gr

Executive Summary

This deliverable describes primarily the process carried out in reach of T3.5 and T3.7, relevant to the development of the xR4DRAMA ontological framework, representation, multi-modal content mapping on semantic entities, Points of Interests management mechanism, and Decision Support System. Furthermore, it reviews the first methodological approach on the reasoning framework and the clustering mechanism.

Based on the requirements structured by WP6 and the dependencies incurring from the interaction with the other WPs (Analysis and fusion of multi-modal data, Platform development, Point Management Mechanism, Decision Support System), the purpose, scope, intended users and uses, and the requirements of the xR4DRAMA ontology were identified. These specifications, along with the modelling understanding from relevant study fields, played an important guidance role for building the first version of the xR4DRAMA ontology that currently comprises modules for capturing the analysis results from the other modules of WP3 (visual analysis, stress level detection and textual analysis). Furthermore, it describes the population process of these incoming data to the repository of the ontology and presents some validation examples. Next, information retrieval mechanisms that are used for text generation and general information displacement, such as emergency numbers, general location description, and related general documents, are developed for the requirements of WP3. A Point of interest management mechanism which creates or updates points of interests, based on the visual and textual analysis content was also developed for the requirements of T3.5. Finally, the integration of a decision support system that assigns task based on events and asserts severity scores on danger zones, was also developed to provide more compact information to first responders and journalists.

Abbreviations and Acronyms

Abox	Assertional Axioms
CQ	Competency Question
DL	Description Logic
ENVO	ENVironment Ontology
GIS	Geographic Information system
KB	Knowledge Base
MEMOn	Modular Environmental Monitoring Ontology
OWl	Web Ontology Language
ORSD	Ontology Requirements Specification Documents
RDF	Resource Description Framework
SOSA	Sensor Observation Sample Actuator
SPARQL	SPARQL Protocol and RDF Query Language
SPIN	SPARQL Inferencing Notation
SSN	Semantic Sensor Network
TBox	Terminological Axioms
WP	Work Package
W3C	World Wide Web Consortium
XML	Extensible Markup Language

Table of Contents

List of Figures

List of Tables

1 INTRODUCTION

This deliverable *D3.11"Semantic representation, fusion and reasoning-based decision support system for situation awareness v2"* focuses on describing an updated view of the xR4DRAMA ontology, the fusion engine, the information retrieval, Points of Interest (POIs) management mechanism, and the Decision Support System (DSS). The ontology, also called "the xR4DRAMA Knowledge Base (KB)", is a knowledge representation model for semantically representing concepts relevant to the project.

The goal of the KB framework within WP3 is to research and develop technologies for semantic content and sensor input modelling, integration, reasoning, and question answering, as well as the fusion of the analyzed data. The models that will be created will constitute the reasoning mechanisms, taking into account the ontology vocabulary and infrastructure for capturing and storing information relevant to the xR4DRAMA application domain, such as: (a) Observation and Events (e.g., data collection of biometric sensors, visual analysis), (b) Spatio-temporal (e.g., highlighted locations and timestamps), (c) Mitigation and response plans in crisis (e.g., first responder teams).

The general architecture of the xR4DRAMA is depicted in Figure 1. The semantic representation repository is a central component in the system's architecture and hosts the xR4DRAMA, with the other components of the system interacting with it through the message broker.

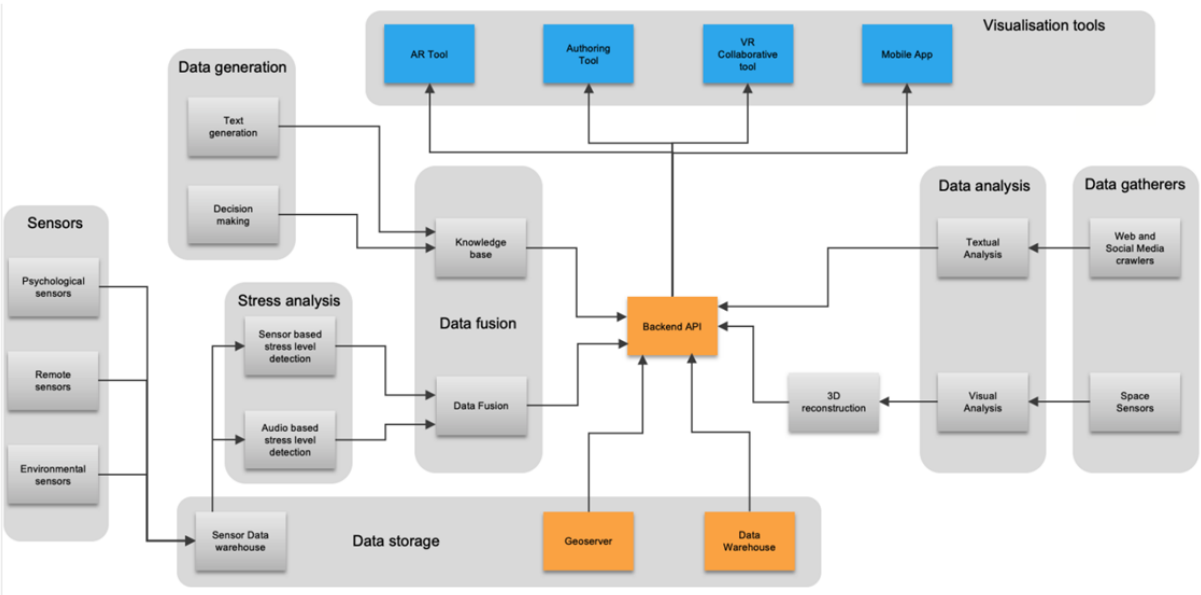

Figure 1: Architecture of the xR4DRAMA project

As for the architecture of the semantic integration component, it is illustrated in Figure 2.

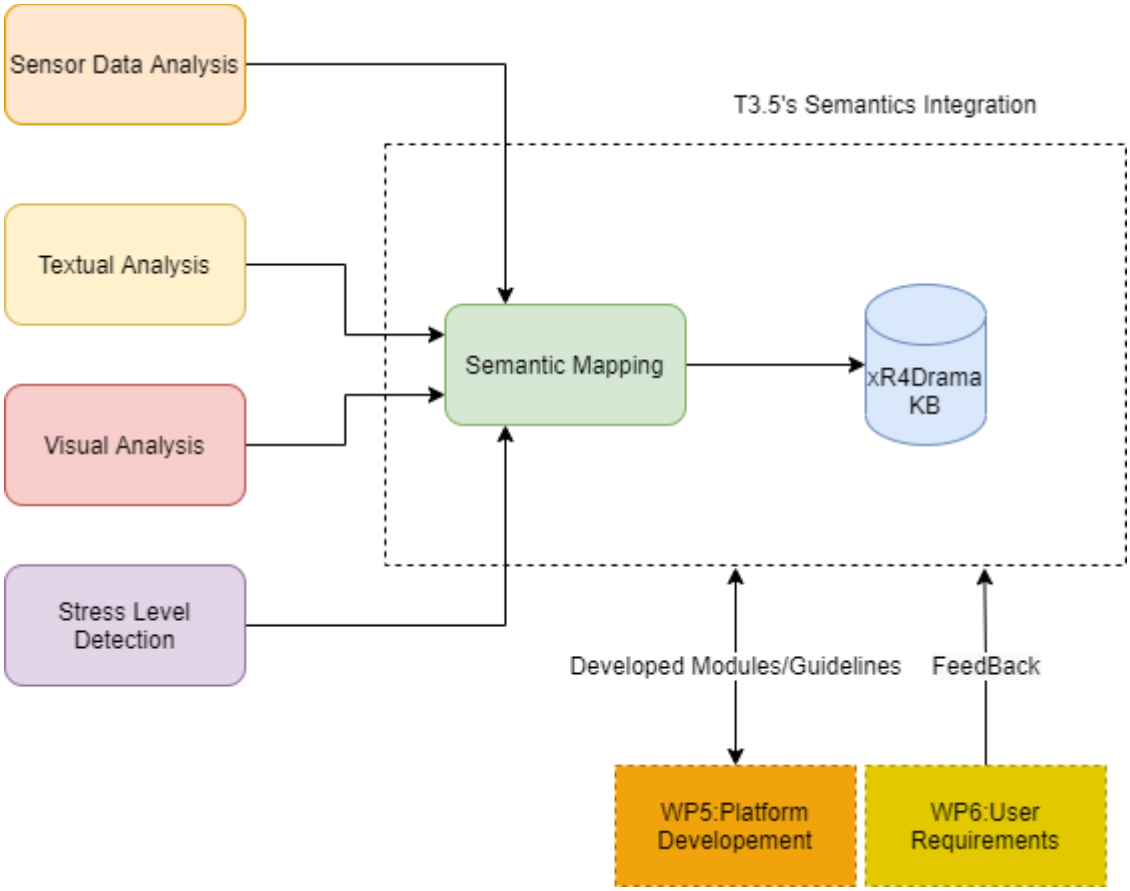

Figure 2: High level architecture of the Semantic Integration

The present deliverable reports on the work process carried out within Task 3.5 and Task 3.7 focusing on the construction of the xR4DRAMA ontology, the multimodal mapping mechanism, the information retrieval mechanism, the POI management mechanism, and the DSS. The idea of the pipeline is that after the multimodal mapping mechanism has received the message from a component, it will map the information into the Knowledge Graph (KG), i.e., populated ontology. Then, when the message arrives from the textual or visual analysis component the POI management mechanism of the xR4DRAMA KG, will create a new POI or update an existing one, based on the information in the message and information from the KG. In the second case, the idea is that the state of a created POI changed, for example a flood has affected more buildings, thus the information in the initially created POI needs update. Also, notice that through the various examples we consider flood disasters, for easier understanding. Figure 3 shows an outline of the pipeline, where each number in the circles shows the order of steps.

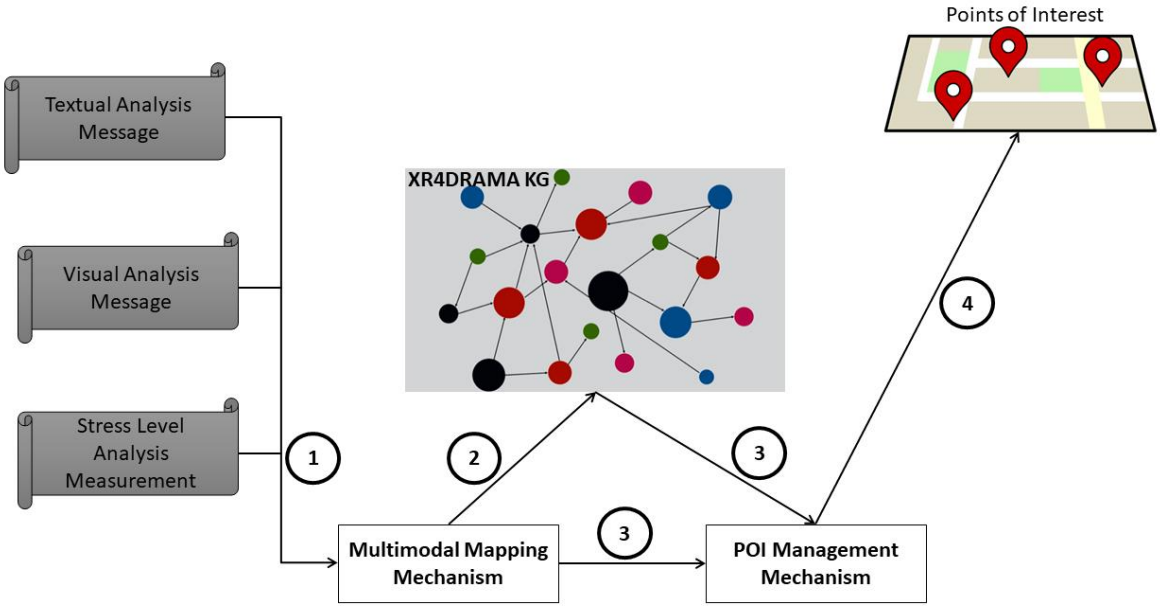

Figure 3: Points of Interest Management Mechanism

The DSS is developed on the POIs management mechanism, as it assigns tasks to first responders, citizens, and journalists, which are attached to the created POIs. Moreover, the DSS also creates danger zones, i.e., bounding boxes, and assigns severity scores, as to how dangerous a location is. Section 2 reviews the relevant state-of-the-art with respect to knowledge representation languages, as well as already existing ontologies addressing project-relevant fields. Section 3 presents the architecture of the KB, how it communicates with the other component, how it processes the information in order to create POIs and danger zones, and how it retrieves information for text generation and general information for locations.

Section 4 reports on the ontology implementation and presents the status of the xR4DRAMA ontology. Section 5 the evaluation for each service and the evaluation of the KB. Section 6 contains some of the fusion theory as well as the semantic reasoning requirements and methodology. Finally, in Section 7, the document is concluded, presenting the conclusions that were drawn and discussing future work and further improvement of the module.

2 STATE OF THE ART

In this section we provide an overview on the state-of-the-art knowledge representation languages, already existing similar domain ontologies addressing relevant data to the xR4DRAMA project. More specifically, we present the core aspects of Description Logic (DL) language (Baade et al 2003) on which the official W3C recommendation for creating and sharing ontologies in the Web (OWL 2) is grounded, some of the OWL 2 categories, as well as relevant rule-based languages. Furthermore, a summary on the representative ontologies that have been proposed in the literature for modelling aspects relevant to the xR4DRAMA domain that fall into WP6's modelling requirements is presented.

2.1 Semantic Web & Knowledge Graphs

Ontology engineering has been widely used as an effective way for modelling specific domain knowledge because they can represent and organise information, context, and relationships more accurately. Additionally, they can be expanded/enriched by merging and combining parts of existing, relative, or not, ontologies into new ones. Ontologies are structures that are used to obtain knowledge regarding a domain of interest. Formally speaking, ontologies are explicit formal specifications of shared conceptualizations (Studer et al., 1998). They show abstract views of the world including the objects, concepts, and other entities that are assumed to exist in some area of interest, their properties and the relationships that hold among them. Their formalization and expressiveness depend on the knowledge representation language used.

The Semantic Web-W3C, which is an extension of the current Web, aims to establish a common framework for sharing and reusing data across heterogeneous sources, ontologies play a fundamental part. The Semantic Web vision is to make the semantics of web resources explicit by attaching to them metadata that describe meaning in a formal, machine-understandable way. Web Ontology Language (OWL) (Deborah and McGuinnes, 2004) has emerged as the official W3C recommendation for creating and sharing ontologies on the Web as the result of the previous effort. In the rest of this section, we present the basics of Description Logic (DL) language, on which OWL semantics are grounded, the different OWL species.

2.1.1 Description Logic

Description Logics is a family of knowledge representation language that may be used for a representation of knowledge of any application domain. This representation pattern is in a structured and formally understandable way. The name DLs derives from two features — the first one is the ability to describe a specific entity with the help of conceptual descriptions; the second one is to provide logic-based semantics.

It is usual for the DLs to include a terminological and an assertional formalism. A set of terminological axioms (TBox) is used to describe labels (or names) for more perplexing descriptions. For example, TBox may contain a description of a concept Mother:

Human ∩ Parent ∪ Mother.

On the other hand, a set of assertional axioms (ABox) is used for description of properties of individuals. For example, we can describe the relationship between two humans, Maria, and her son Alex:

hasChild(Maria, Alex)

DLs offer a reliable tool to deduce implicit knowledge from the explicitly defined knowledge with the help of TBox and ABox. The DLs provide well-defined semantics and powerful reasoning tools. For many years, there was a mismatch between the expressivity of DLs and the efficiency of reasoning. In other words, if a user wants to use a DLs, then he needs to establish a trade-off between the expressivity of DLs and the complexity of their inference capability. It means it is needed to restrict DL appropriately.

2.1.2 Knowledge Representation

The OWL belongs to the Semantic web, which has been created to represent plentiful and complex knowledge about things, groups of things and relations between things. OWL can be described as computational logic-based language. For this purpose, OWL can be easier for machines to automatically process and integrate information available on the Web.

OWL uses RDF's XML syntax (RDF/XML). OWL has adopted several features of RDF/RDFS meaning of classes and properties and those language primitives are beneficial to overall expressiveness. On the other hand, RDF and RDFS have very voluminous modelling concepts such as rdf:Property and rdfs:Class. Thus, RDF and RDFS may be restricted when a trade-off between expressive power and efficient reasoning must be established. There are three main kinds of OWL because of the trade-off mentioned above.

Different sub-languages are described in the following list:

- OWL Full: this kind of OWL represents the entire OWL language. This kind also offers the possibility to combine OWL primitives and RDF and RDFS. Moreover, the meaning of predefined primitives may be changed. OWL Full provides full compatibility with RDF, i.e., every valid RDF document is also valid OWL Full document. On the other hand, there is a possibility for the ontologies developed in OWL Full to be undecidable.

- OWL DL: this kind of OWL, where DL stands for Description Logic, restricts the application of constructors from OWL and RDF. The restrictions include: (1) Vocabulary partitioning, (2) Resources are allowed to be only one of specific type, i.e., a class, a datatype property, an object property, an individual, etc. Strictly speaking, a property cannot be a datatype property and at the same time object property and vice versa. The efficient reasoning is secured because of: (a) explicit typing of resources, (b) no transitive cardinality restrictions, (c) restricted anonymous classes. Furthermore, compatibility with RDF is lost. On the other hand, every valid OWL DL document is a valid RDF document.

- OWL Lite: is the last version that represents a restriction of OWL DL. The restrictions are, for example, excluding enumerated classes, disjointedness of classes, and cardinality (except the values 0 or 1).

2.1.3 Semantic Querying for Reasoning

As it was aforementioned, DLs, as well as OWL, exchange some expressiveness for more competent reasoning. The tree-model property is one such example. It conditions the tree-shape structure of models, ensuring decidability, but at the same time it severely restricts the way variables and quantifiers can be used, dictating that a quantified variable must occur in a property predicate along with the free variable. Consequently, it is not possible to describe classes whose instances are related to an anonymous individual through different property paths. To overcome OWL's limited relational expressivity and modelling shortcomings, the research body came up with the integration of rules with OWL.

The first step toward this was SPARQL, a language recommended by the W3C for extracting and updating information in RDF graphs. It is characterized by expressiveness with the ability to describe complex interactions and relationships between entities in a knowledge graph. The semantics and multiplicity of the SPARQL language have been reviewed in detail theoretically, showing that SPARQL algebra has the same expressive power as relational algebra (Perez et al., 2006). Even though SPARQL is mainly used as query language for RDF, by using the CONSTRUCT graph pattern, it can define SPARQL rules that by combining existing RDF graphs into larger ones can create new RDF triples. These rules are defined in the interpretation layer in terms of a CONSTRUCT and a WHERE clause: the former defines the graph patterns, i.e., the set of triple patterns that should be added to the underlying RDF graph upon the successful pattern matching of the graphs in the WHERE clause. The SPARQL Inferencing Notation (SPIN) (Knublauch et al., 2011) helps with the establishment of an easier expression and execution of SPARQL rules on top of RDF graphs. In SPIN, SPARQL queries can be stored as RDF triples together with any similar domain model, enabling the linkage of RDF resources with the associated SPARQL queries, as well as sharing and reusing them. SPIN supports the definition of SPARQL inference rules that can be used to derive new RDF statements from existing ones through rule application. A newer standard that has been developed as a tool to define structural constraints on RDF charts is Shapes Constraint Language (SHACL). SHACL consists of two parts: (1) a core that elaborates RDF vocabulary for the definition of shapes and variables and (2) SHACL-SPARQL, which is a mechanism for expanding the SPARQL.

2.1.4 Knowledge Graphs

The scope of this subsection is to present the state-of-the-art ontologies that can be used for modelling aspects relevant to the xR4DRAMA's domain of application. According to the xR4DRAMA ontological requirements, which will be reviewed in the following section, we have categorized the relevant ontologies into four domains. First, the ones that can be used to model events and observations. Next there are the crisis management ontologies (modelling risks and mitigation) followed by the disaster ontology and finally the ontologies for general purposes; temporal and geospatial. We should say at this point that the purpose of this section is not to provide a complete list of the ontological structures related to the xR4DRAMA's domain, but to highlight on design concepts and entities that have been proposed or used in systems for modelling and conceptualization.

Observation and Events: The mapping of sensors and their observations, properties and features of interest has been in the centre of many research approaches. Into this, the

dominant ontologies are the Semantic Sensor Network (SSN) (Compton et al., 2012) and Sensor Observation Sample Actuator (SOSA) (Janowicz et al., 2019). They have been applied in various use cases, applications and scenarios, including satellite imagery, large-scale scientific monitoring, industrial and household infrastructures, social sensing, citizen science, observation-driven ontology engineering, and the Web of Things.

The ontology that was studied is Modular Environmental Monitoring Ontology (MEMOn) (Masmoudi et al., 2019). MEMOn is based on other ontologies, namely the Basic Formal Ontology (BFO), the ENVironment Ontology (ENVO) (Buttigieg et al., 2013), the Semantic Sensor Network Ontology (SSN) and the Common Core Ontologies (CCO). In Figure 4 it is shown that the ontology offers eight main modules (Disaster, Temporal, Environmental material and process, Sensor, Observation Geospatial and Infrastructure) covering more aspects than the ontologies to represent different emergency incidents.

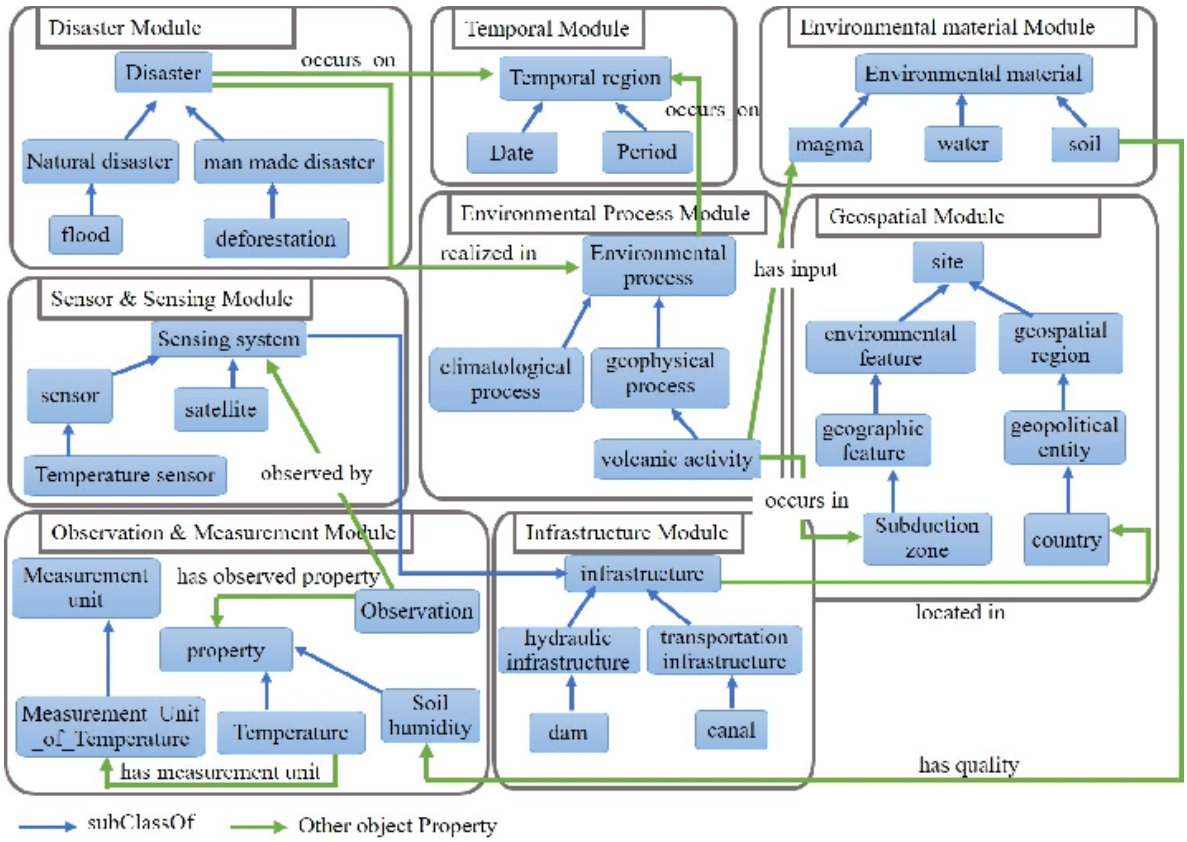

Figure 4: MEMOn Ontology

Multimodal Context: Multimedia Object Description ontology (Choudhury et al 2008) in Figure 5 is aimed to integrate the multimedia objects on the web with other information objects to give an interlinked and integrated view of the user information needs. This is only possible when we model not only the media object used to represent the content but also the content abstracting the higher semantic concepts of specific domain. The basic entities that are described through this ontology are:

- Multimedia Objects: with four subclasses, like Video, Audio, Image, Segment.

- Feature: to describe the audio and visual characteristics of the data.
- VisualConcept: a class for describing semantic concepts in a higher level. They can be simple (car, human) or complex (flood, explosion).
- Event: describes the semantic content in terms of events, which is interplay between objects and actions or other sub-events. Event detection is a challenging issue for multimedia processing and retrieval community.
- Location: two types of location, as depicted in the media, such as City, and spatial location, such as coordinates.
- Segment: is the broader class for both spatial segments and temporal segments, such as image region, frame and shots, audio segments.

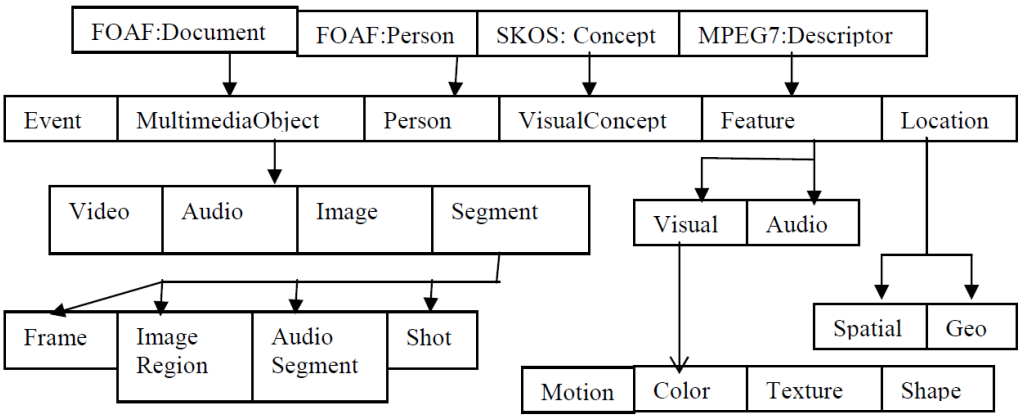

Figure 5: Multimedia Ontology

Crisis and Disaster Management: The construction of context information for the disaster management ontology (Hoill and Chung, 2015) is divided in three different sections. The external context information, which contains the environment, location, and equipment information. The internal context information, which takes inputs from users and their personal and position information. And finally, the service context information, which is divided into guidelines and location service information. Figure 6 shows the person-based relations with other classes, and it consists of internal ontology and external ontology. Therefore, the information for personalized service is drawn and offered through service state, user position, device operation, context information about disaster, and user environment information.

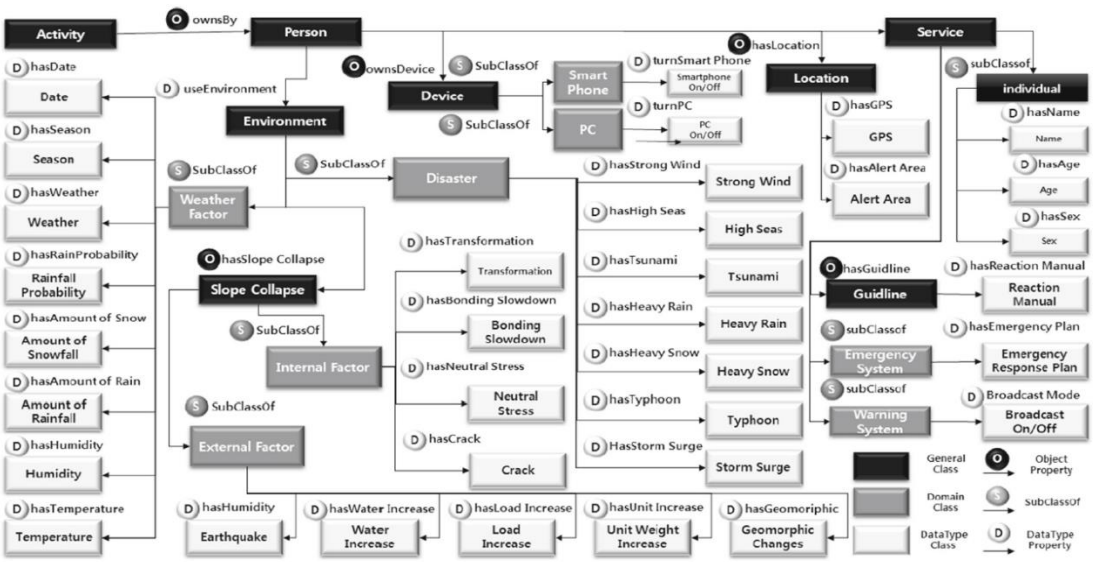

Figure 6: Disaster Management Ontology

As for a more specific example of a disaster ontology, we mention here the Flooding Knowledge Graph (Son et al., 2021). In Figure 7 it is illustrated the structure of the graph that utilizes concepts from different sources (e.g., Wikidata[1], DBpedia[2]) to describe suitably entities like "cause" and "effect". Also, this ontology is a good example to present the interlinking with other knowledge graphs as a high compatibility feature.

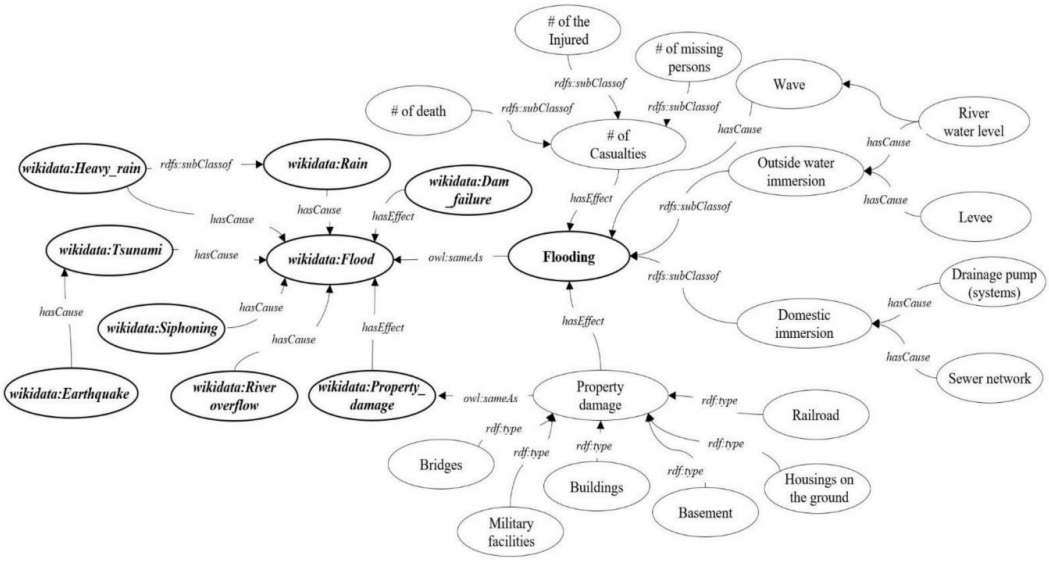

Figure 7: Flooding Ontology

[1] https://www.wikidata.org/
[2] https://www.dbpedia.org/

2.2 Related Knowledge Graphs based Frameworks

In this sub section we will give a small overview of KGs for Disaster Management, KG for Media Planning, and some time and geospatial related studies.

2.2.1 Knowledge Graphs for Disaster Management

The first category of knowledge graphs that can be considered close to the xR4DRAMA KG, are KGs for destructions. It is worth noticing that this area of KGs is not very rich, as there were just a handful of studies related to KGs for destructions. Some interesting cases are (Li and Li, 2013) and (Chou et al., 2010). In the first one, the authors present a KG for semantic representation of textual information, whose main aspect is the multi-document summarization for disaster management. However, in the second study, the authors present similar KGs (i.e., multi-document summarization for disaster management), but for information existing in various websites for disaster management. The difference between the previous studies and the xR4DRAMA KG, lies mostly in the aspect that xR4DRAMA KG can represent knowledge for visual messages and stress-level measurements, and can also create POIs which can help in real-life scenarios.
Next, the study of (Murgante et al., 2009) presents a KG for seismic risk domain. The authors develop a KG which offers an improvement of semantic interoperability, in order to decrease the economic and social costs, deriving from seismic events, which by extension can lead to a prevention strategy to reduce damages. The COVID-19 KG (Wise et al., 2020), was also constructed in order to improve the semantic interoperability of the literature about the current pandemic. Nevertheless, the xR4DRAMA KG is more general than the aforementioned studies, as it is not restricted to only one type of disaster. The area of KGs for disaster management is richer than the area of KGs for destructions, some papers that present a blueprint of what a KG for disaster management should contain are presented in (Werder, 2007) and (Xu and Zlatanova, 2007), (Klien et al., 2006). In the last two the focus is mostly on geospatial information about a disaster, while the first one is more general. The difference between these studies and xR4DRAMA KG, is that they remain at a theoretical level while we offer a complete KG with a POI management mechanism.

In (Zhang et al., 2020), (Purohit et al., 2019), the authors present a deep learning model that can generate a KG for disaster management, but as most data driven models it is restricted upon the data that is trained. This means that if a new case needs to be inferred, for instance a different type of disaster, new classifiers need to be trained. Comparing this to the xR4DRAMA KG which is not restricted to the information existing in some datasets, shows that our KG might be more general than these models. Close to our study is (Moreira et al., 2015), where the authors present a KG for disaster management, but they do not include a POI management mechanism, for accessing the information in the KG. Similar is the case of (Babitski et al., 2011), as there is no POI management mechanism. One can take a more detailed view for the KGs about disasters and disaster management by reading the survey of (Mazimwe et al., 2021).

2.2.2 Knowledge Graphs for Media Planning

The area of KGs for media planning is not very rich, as only a handful of studies can be classified in this area. For instance, the studies of Opdahl et al. (Opdahl et al., 2016) and Berven et al. (Berven et al., 2018) are two similar studies which present mechanisms for media planning. The basic concept for both these studies is that they offer a news extraction mechanism which based on the semantics of a KG, will extract related posts from social web sites, and other well-known media houses, about an event. This synergy of news extraction, and subsequently representation of knowledge about an event, is set to help a journalist to see what parts of the event have been covered, and what are the restrictions for accessing a location to cover the event. The difference with the xR4DRAMA KG lies mostly in the POI management mechanism, as we offer the most crucial information about an event in a POI, and therefore make it more easily digestible for the journalists, while (Opdahl et al., 2016) and (Berven et al., 2018) return information in the form of text which can be more time consuming for an individual to process.

The area of media planning in disasters based on KGs is also not so rich, as to the best of our knowledge, only a few studies exist in this area. KG in media is mostly used for fake news detection (Pan et al., 2018) and building event-centric news (Rospocher et al., 2016; Tang and al., 2019). Some exceptions are (Wang and Hou, 2018) and (Ni et al., 2019). In the former, the authors propose a method to construct a KG for disaster news based on an address tree. Address Trees are tree structures which analyze an address having as root the broader region. For example, home address → town district → town, is a small address tree. In the latter, the authors present a data driven model which generates storylines from huge amount of web information and proposes a KG-based disaster storyline generating framework. For the work of (Wang and Hou, 2018), comparing to xR4DRAMA KG there is not a mechanism for creating POIs, and the indication for the location is given in string descriptions which can be obscure in some cases, while xR4DRAMA KG represents locations with coordinates.

The study of (Rospocher et al., 2016) presents methods and tools to automatically build KGs from news articles. As news articles describe changes in the world through the events they report, an approach is presented to create event centric KGs using state-of-the-art natural language processing and semantic web techniques. Such event centric KGs capture long-term developments and histories on hundreds of thousands of entities and are complementary to the static encyclopedic information in traditional knowledge graphs. Even though these two studies might not solve the same problem with xR4DRAMA KG, the crucial information can be accessed through sophisticated SPARQL queries, which might not be user-friendly even with an UI. On the other hand, xR4DRAMA through its POI management mechanism serves the crucial information about an event, with a POI, which is more easily understandable by an individual.

2.2.3 Time and Geospatial Data

In semantic web there are two standard ontologies of temporal concepts, OWL-Time (Hobbs and Pan, 2004) and time-entry (Pan et al 2004). They both provide similar vocabularies for expressing facts about temporal intervals and instants, while time-entry also includes the concept of an event. In addition, the ontologies include classes and relations for expressing time intervals and instants in clock and calendar terms. Both include the concept of a time zone, and a separate global time zone recourse in owl is available.

The importance of the geospatial data (e.g., locations, distances, coordinates) and their semantic representation is recognised by the research, because they offer solid methods for retrieving information that are used in several GIS applications. There are many geographical ontologies that are used to express semantically geographical and spatial information. One of the most prominent of them is the GeoSPARQL. The later defines an RDF/OWL vocabulary for representing the aforementioned information and elaborates them with the use of a query language with powerful rules and functions, which allow precise semantic reasoning.

3 PREVIOUS WORK

In this section we describe the roadmap and the methodologies followed to collect the modelling and reasoning requirements, as well as a description of the results of this approach. Additionally, there is an effort on the association of the modelling and reasoning requirements with technical & user requirements.

3.1 Methodology

The methodology that we followed to formulate modelling and reasoning requirements for the xR4DRAMA KB can be visualized with the use of structural blocks of developing actions. In Figure 8 there is a high-level review of these milestones.

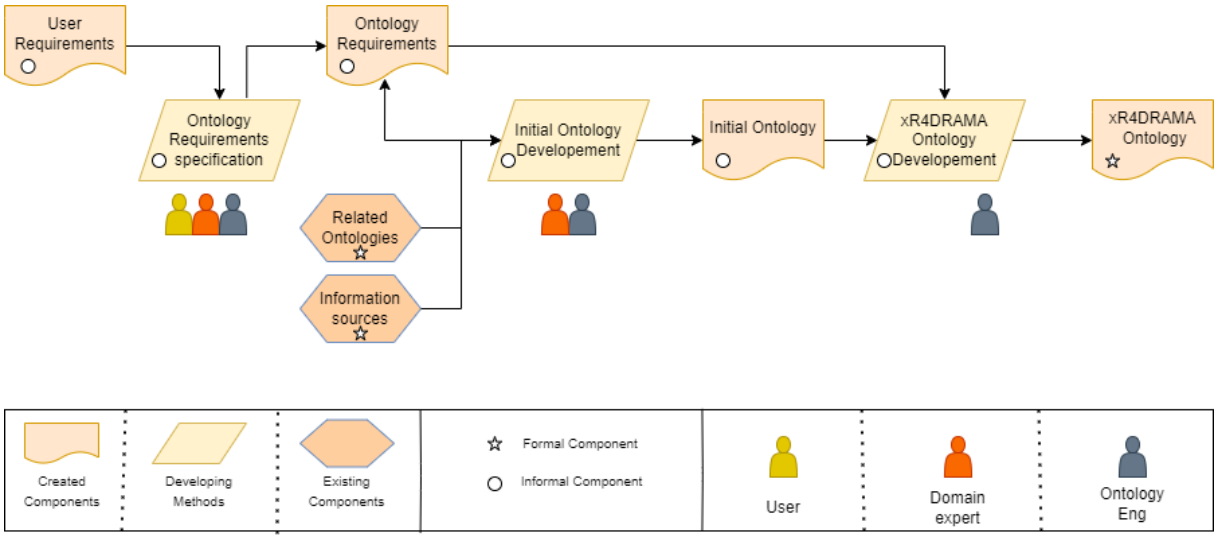

Figure 8: Methodology for requirements elicitation

The process that was followed can be divided into three major stages with several possible inputs and outputs.

1.The first stage is focused on ontology requirements specification and the retrieval of ontology requirements specification documents (ORSD, described in the following section). In this stage the role of end users is of great importance. They will provide insights regarding the user requirements. Additionally, domain experts will help understand the use cases and find the optimal matching with the ontology requirements. Finally, ontology engineers have a more consulting role in this stage regarding the process execution.

2. The second stage, after the acquisition of ontology requirements, involves the development of an initial ontology making a good use of related ontologies of the same domain, and information from several outputs of the xR4DRAMA system, which have filtered with the results of the first stage. The role of the ontology engineers here is major, whilst he translates the domain experts' findings into a machine interpretable format.

3. The third stage contains the expansion of the initial ontology with the use of more advanced design patterns and further specification of the incoming information, with the use of the OWL to finalize the xR4DRAMA ontology.

3.2 Ontology Requirements Specification

As we mentioned before, the important role in the first stage of the methodology that followed, was the Ontology Requirements Specification Document (ORSD) (Suarez-Figueroa et al., 2009). This is a template-based report in which we determine which are the domain and the scope of the ontology. Additionally, this document helps us to specify why the ontology is needed in the project, what are the intended uses, who are the end users, what the ontology should fulfil, and the verification, grouping and prioritization of requirements.

3.2.1 Ontology Requirements Specification

The template of a ORSD contains the following fields where you can find information regarding the purpose, scope, implementation language, intended end-users, intended uses, requirements, and pre-glossary of terms of the ontology that is being built:

- Purpose: The main general goal of the ontology/main function or role that the ontology should have.
- Scope: The coverage and the number of details that the ontology should contain.
- Implementation Language: The formal language that the ontology should have.
- Intended End-users: The intended end-users expected to need the ontology.
- Intended uses: The intended uses expected for the ontology.
- Ontology requirements:
    - Non-functional requirements: The general requirements or aspects that the ontology should fulfil, including optionally priorities for each requirement
    - Functional Requirements: Groups of Competency Questions (CQ): The content specific requirements that the ontology should fulfil in the form of groups of competency questions and their answers, including optional priorities for each group and for each competency questions (Noy and Mcguinness, 2001).
- Pre-glossary of Terms:
    - Terms from Competency Questions: Items that included in the competency questions and their frequencies.
    - Terms from Answers: The terms that included in the answers and their frequencies.
    - Objects: The objects that included in the competency questions and their answers.

3.2.2 xR4DRAMA OSRD

The xR4DRAMA ORSD is based on the use cases scenarios and requirements laid out in D6.1 “Pilot use cases and initial user requirements” and in D6.2 “Final user requirements”. Additional feedback and clarifications have been elicited through iterative cycles of communication with WP3, WP4, and WP5 that extended equally and were qualified to provide supplementary analysed input that ultimately came to further refined and unambiguous requirements. Therefore, the previous process results in the ORSD that reflects the ontology requirements as pertinent to the status of the xR4DRAMA system. It is possible that some

revisions and extensions will need to be carried out as the system functionalities evolve. Table 1 constitutes the xR4DRAMA ORSD.

1	**Purpose**
	As the purpose of the xR4DRAMA semantic representation framework we can define the structures and the vocabularies that are used to capture the analysis results coming from other components. The system needs the ontology to secure interoperability and reusability between the individual modalities and to support, together with inference rules, personalised (based on the users) interpretation services. The KB will be the crossroad between the sensor data inputs and the backend.
2	**Scope**
	The ontology has to focus just on the following aspects: • Representation of the analysed data from Stress Detection sensors. • Representation of the analysed data from Visual Analysis tools. • Representation of the analysed data from Text Analysis tools. • Representation of the georeferenced data. • Representation of the analysed data from mobile application.
3	**Implementation language**
	The ontology will be implemented in OWL 2, presented in previous section, the officially recommended language by W3C for knowledge representation in the Semantic Web.
4	**Intended End-Users**
	• **PUC1: Authoring Tool** Displaying Messages to the authoring tool regarding an emergency situation, so the specific mitigation actions to take place.
5	**Ontology Requirements: Functional Requirements - CQs**
1. Analysed Data 1.1. What is the severity of the observation [X]?	

1.2. What is the risk level of the observation [X]?
1.3. Which is the emergency in the observation [X]?
1.4. What is the detection/creation time of the observation [X]?
1.5. Which is the area in the observation [X]?
1.6. What is the probability of the area in observation [X]?
1.7. Which is the Stress level of the between time intervals [t_1]-[t_2]?
1.8. Which is the objects found in video[X]?
1.9. Which is the simmoid in video[X]?
1.10. What is the multimedia type used in observation [X]
1.11. Which is the most/least risky observation?
1.12. Which buildings where detected between time intervals [t_1]-[t_2]?
1.13. What is the probability of a detected building/object?
1.14. Which observations occurred after time [t_1]?
1.15. How many people are in danger between time intervals [t_1]-[t_2]?
1.16. How many vehicles are in danger between time intervals [t_1]-[t_2]?
1.17. How many open areas are between time intervals [t_1]-[t_2]?

2. Geospatial Data
2.1. What is the location of the FR[X]?
2.2. What is the location of the observation[X]?
2.3. What is the location of the citizen-user[X]?
2.4. Which observation has location[X]?
2.5. What is the project location[X]?
2.6. How many citizens[X] are close to the specific location?
2.7. How many FR[X] are close to the specific location?
2.8. What is the location of the safe area ?
2.9. What is the location of the risk area?
2.10. Which are the coordinates of a safe route?

3. Project Data
3.1. What is observations related to the project[X]?
3.2. What is the id of the project [X]?
3.3. What is the RiskReport of the project [X]?
3.4. How many citizens are in the specific project[X]?
3.5. How many FR are in the specific project[X]?

Table 1: xR4DRAMA OSRD

3.3 Multimodal Mapping Mechanism

Our multimodal mapping mechanism refers to the mechanism that integrates into the ontology the messages received from the visual, textual and stress level analysis components. More specifically, the messages received from the components are passed to the mechanism as JSON-structured messages and are translated into Turtle format. Turtle is a more fine-grained format for Resource Description Format (RDF). Below we provide one visual analysis message, one stress level analysis message, and two textual analysis messages one for each use case, as the textual analysis messages have different structure comparing to the other two components.

Visual Analysis Data: The following example was used as an input in the mapping service of the conversion from JSON to RDF. The visual analysis module sends its data in the following API *http://xr4drama.iti.gr:8090/rest_api/webresources/secured/population/{visuals}*.

Visual Analysis Message


```
{
  "header": {
    "timestamp": " 2020-03-24T13:02:08.69",
    "sender": "Visual Analysis",
    "entity": "video",
    "simmoid": "1408037303193309186"
  },
  "shotInfo": [
    {
      "shotIdx": 0,
      "startFrame": "0",
      "endFrame": "203",
      "objectsFound": [
        {
          "type": "wall",
          "probability": 0.56
        },
        {
          "type": "person",
          "probability": 0.82
        }
      ],
      "peopleInDanger": 0,
      "vehiclesInDanger": 0,
      "riverOvertop": false,
      "infraInDanger": ["house", "building"],
      "objectsInDanger": ["bench", "computer"],
      "animalsInDanger": 0,
      "area": "berth",
      "areaProb": 0.3972055555555555,
      "outdoor": false,
      "emergencyType": "none",
      "emergencyProb": 1,
      "category": "Transportation",
      "subcategory": "Taxi",
      "coordinate": [35.523, 24.075]
    }
  ]
}
```


Stress Level Data: As with the visual analysis, we have the stress level output analysis in the form of the following JSON. This module sends its data in the following API *http://xr4drama.iti.gr:8090/rest_api/webresources/secured/population/{stress}*.

Stress Level Analysis Message


```
{
  "Latitude": 40.5993542,
  "Longitude": 22.9756221,
  "Probability": "NULL",
  "Stress_Level": "43.47095787525177",
  "Timestamp": "24-11-2021 14:33:39",
  "User_ID": "admin"
}
```

Textual Analysis Data: The textual analysis component gives two different JSON files for mapping into the KB, each one for each use case. Thus, we provide two examples one for use case 1 (Example 1) and one for use case 2 (Example 2). This module sends its data in the following API *http://xr4drama.iti.gr:8090/rest_api/webresources/secured/population/text*.

Textual Analysis Message Use Case 1


```
{
  "meta": {
    "entity": "citizen_report",
    "project_id": "41",
    "id": "simmo)id",
    "sourceText": "some text",
     "location":{
           "radius": 0.0,
           "inferred": "true",
           "coordinates": [37.985, 23.745]
     }
  },
  "data":
    {
      "timeReference": 0,
      "tags": ["Human in Danger", "Building in Danger"],
      "endFrame": "203",
      "objectsFound": [
        {
          "label": "obstruction",
          "agents": [],
         "affected_objects":[
            {"name": "person",
            "quantity": "some"},
            {"name": "car",
            "quantity": "3"}

           ],
             "location": "house"
        }
      ],
      "type": "incident",
      "labels": ["Human in Danger", "Building in Danger"],
     "category": "Disaster Management",
     "subcategory": "Civil Protection Distribution Places"
    }

}
```

Textual Analysis Message Use Case 2


```
{
  "meta": {
    "entity": "citizen_report",
    "project_id": "41",
    "id": "simmo)id",
    "sourceText": "some text",
     "location":{
           "radius": 0.0,
           "inferred": "true",
           "coordinates": [37.985, 23.745]
     }
  },
  "data":
    {
      "utilities": [
        {
          "type": "Power supply",
          "qualities": [],
         "quantity": "1",
         "relative_position": null,
         "location": "table"
        }
      ],
      "type": "logistics",
      "coordinates": [37.977217, 23.730052],
     "category_name": "Juice Bar",
     "name": "JOIN juice bars"
    }

}
```

4 xR4DRAMA ARCHITECTURE

In this chapter we present a thorough analysis of the architecture related to the KB. In more detail we describe: (i) how the KB receives and integrates the messages from the textual, visual and stress level analysis component, (ii) how it retrieves information, i.e., information for textual generation and information which contains the general description of the location, the legislation for recording and drone usage, and emergency numbers, (iii) how the KB creates, or updates POIs based on the messages receive from the textual and visual analysis components, and (iv) how the KB communicates with the DSS system in order to create tasks for the POIs, and assign severity scores to danger zones.

4.1 System Architecture

The architecture of the KB has two main pipelines which involve the functions offered by the KB. In the first, one the KB receives messages from the visual, textual, and stress level analysis components it maps the information component from these messages provided by the multimodal mapping mechanism. Subsequently, based on the most important information contained in the messages it will create or update a POI (see subsection 4.2). Figure 9 depicts the previous pipeline.

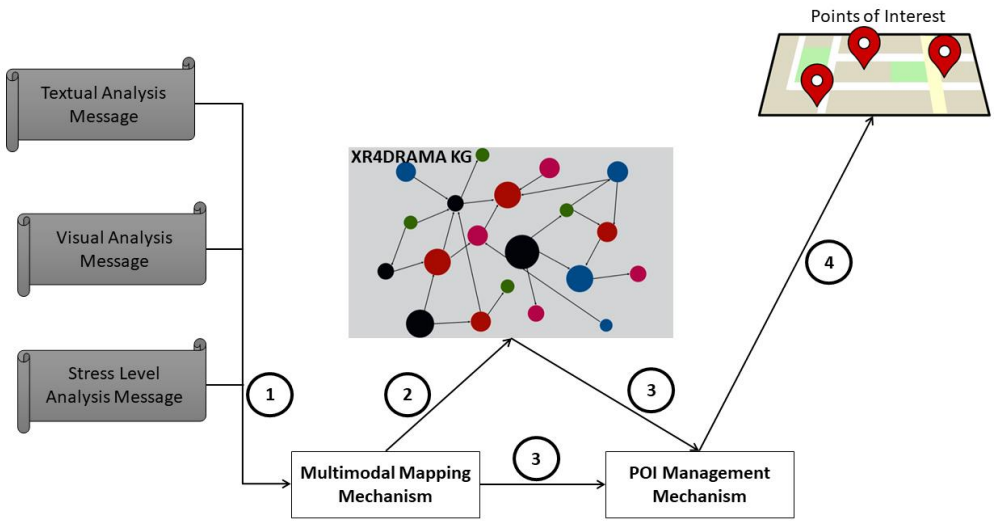

Figure 9: Multimodal Mapping and POI Management Mechanisms

Notice that POIs are created or updated only based on the information contained in the visual and textual analysis messages. The reason for that is because the stress level analysis messages contain only the stress measurement of the first responders (i.e., civil protection workers, military personnel, journalists, etc.), which is not very crucial information for a first responder, who in disaster management situation needs information like: which is the state of the location, which object(s) got affected, what infrastructures were affected, and how many persons are in danger.

Regarding the nature of data, the use of a semantic KG was necessary to meet the project's needs due to the system's multimodality, diversity, and need for homogeneity and fusion.

There is an underlying data storage facility for this purpose, thus the KG is not in charge of archiving and storing raw data files. Instead, the KG stores raw data metadata, analysis findings, and other material with semantic value that might be mapped and combined with other candidates to construct the knowledge base. The main types of information that needed to be recorded in the KG were: (i) general data about virtual reality experiments, (ii) visual analysis results from images and videos, and (iii) textual analysis results derived from online retrieved content.

The visual analysis component is an off-the-shelf tool, which is the result of some of our previous work (Batziou et al., 2023). In more detail, the visual analysis is a computer vision mechanism that can identify objects in an image or video and classify the image into a specific category which is called verge of the image. The Verge classifier was used to assign the photos (or video frames) to one or more classes depending on context (Andreadis et al., 2020), while the model conducting semantic segmentation (Qiu et al., 2021) on images was trained to extract semantic labels and percentages per pixel on images.

Since it was decided not to deep copy structures from a SOLR instance, which do not serve any requirements, only a small number of the generated assets were chosen to be included in the KG. These assets include the text itself, the sentences that make up the text, and the named entity relationships that may be found there.

On the other hand, regarding the pipeline of the KB which communicates with DSS component, there are two functionalities. In the first one, the DSS sends information about tasks which need to be performed by first responders and citizens in a disaster management scenario, and the KB will attach them to specific POIs. These messages are triggered by external sensors which measure water levels. In the second one, the DSS sends a severity score which will be attached to an existing danger zone or a new danger zone which needs to be created. Figure 10 depicts how based on the messages of the DSS it attaches tasks to POIs or creates/updates danger zones based on severity scores.

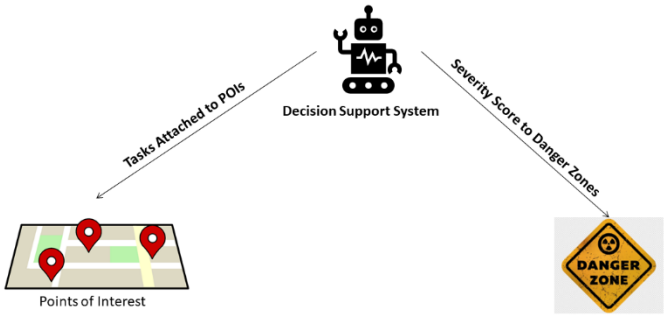

Figure 10: Task and Danger Zone Creation from DSS

4.2 Points-of-Interests Creation or Update

When covering a recording of a documentary in a remote location, unknown to the media production team, a journalist should have access to geospatial data that gives details on the location they will be visiting. Also, for an individual in a disaster management situation it is also important to access geospatial data that contains information about the location that suffered the destruction. The information will enable him or her to accurately and cost effectively setup the production. For this reason, we offer a tool that allows users to add or update Points of-Interest (POIs). We use the term POI management mechanism to refer to the process for creating and updating POIs, throughout the paper. A POI, according to the official definition, is a particular place or location point on a map that a user would find useful or interesting. In our situation, POIs also comprise geospatial data that includes details from user provided videos, photos, and text messages on the condition of a location. As a result, POIs contain data that can assist journalists in actual situations involving media preparation.

In order to make it easier for first responders, journalists and media organizations to complete a remote production mission, POIs aim to create some points in a region (i.e., pins on a map) that convey vital geographical information. Any user can add or modify POIs using the AR app. The user can either provide a picture or video that the visual analysis component analysis, and some of the important data in the image or video is then passed to a new or existing POI (the pipeline for a textual message is similar).

It is simple to comprehend why a POI would need to be formed: if an event had taken place and there were none already present in the region. The updating of POIs, on the other hand, takes place when there are already POIs in the region and part of the information in them needs to be updated since the event's state has changed. For instance, the area has become crowded. The data from a visual message that is sent to a POI during creation (see Table 2) or updating (see Table 3) is shown below. The endpoint for the POI creation/update based on visual analysis messages is: *http://xr4drama.iti.gr:8090/xr4d_Retrieval/webapi/textualPOI*.

Label	Value	Example
category	string	Education
subcategory	string	University
current_user	string	Alexandros Vassiliades
objectsDetected	list of strings	[car, cabinet]
sceneRecognition	string	theatre
type	string	point
coordinates	list of floats	[11.55,45.54]

Table 2: Information passed from a visual message to a POI when created

Label	Value	Example
objectsDetected	list of strings	[car, cabinet]
sceneRecognition	string	theatre

Table 3: Information passed from a visual message to a POI when updated

One can notice that when a POI is created the information passed from the visual analysis messages are: (i) what objects have been recognized, and (ii) the label of the scene (i.e., the

scene is the verge classification of the image see subsection 3.1). Since some of the previous data may be dynamic, the POI will still be constructed even if some are absent. The category and subcategory characterize the area, which was recognized, in our running example an education area, and more particularly, a university was recognized. In addition, the current user is the name of the user who sent the message. Finally, the coordinates are also given in the format: [longitude, latitude]. The current user, the category, the subcategory, and the location are required pieces of information. However, only a limited amount of information can be altered if a POI already exists. The data which can be updated are: (i) and (ii). We also analyse the information from a textual message that is passed to a POI when is created (see Table 4) or updated (see Table 5).

Label	Value	Example
category	string	Education
subcategory	string	University
current_user	string	Alexandros Vassiliades
sourceText	string	the amphitheatre suffered a flood
objectsDetected	list of strings	[car, cabinet]
label	string	amphitheatre
type	string	point
coordinates	list of floats	[11.55,45.54]

Table 4: Information passed from a textual message to a POI when created

Label	Value	Example
sourceText	string	the amphitheatre suffered a flood
objectsDetected	list of strings	[car, cabinet]
label	String	amphitheatre

Table 5: Information passed from a textual message to a POI when updated

Similarly, when a POI is created the information passed from the textual analysis messages are: (i) which are the detected objects, (ii) an auxiliary label that characterizes the location, and (iii) the source text of the textual message. The previous data can be dynamic, meaning that even if some are missing the POI will still be created. The necessary data is the current user, the category, the subcategory, and the coordinates. On the other hand, if a POI already exists only some information can be updated. The data which can be updated are: (i-iii). The endpoint for the POI creation/update based on visual analysis messages is: *http://xr4drama.iti.gr:8090/xr4d_Retrieval/webapi/viusalPOI*.

4.3 Decision Support System

One of the main goals of the Decision Support System (DSS) is to generate the Severity Level (SL) of a POI fusing the information that is available to the KB, either via the Textual or the Visual Analysis outcomes. The severity score is then used from the KB to create or update the

situational picture of a danger zone containing that POI. Two different approaches are used in order to assess the current severity of a situation, depending on the type of the analysis message DSS receives from the KB (Textual or Visual).

For the Textual Analysis messages, a rule-based approach was used. When a textual message is generated, KB communicates that information to DSS in real time. The following message is an example of such a textual message.

KB message to DSS for textual analysis severity assessment


```
{
   "affected_objects":[
      {
         "No":"1",
         "Class":"car"
      }
   ],
   "dangerInfo":{
      "severity":"low",
      "latitude":37.988,
      "created_at":"2022-11-22T10:05:20.852120+00:00",
      "id":"28",
      "longitude":23.75
   },
   "location":"flood",
   "geometry":[
      23.75,
      37.988
   ],
   "id":"28",
   "type":"Textual",
   "category":"Disaster Management",
   "subcategory":"Civil Protection Distribution Places",
   "Timestamp":"2022.11.22.12.05.23"
}
```


If the event is categorized as a “flood” then DSS is assigning a weight value to each affected object that is listed, according to its general “class”. Heuristic rules were used to generate the classes and their respective weigh, that can be seen in the table below (Table 6).

Class	Weight
People	0.9
Animals	0.85
Vehicles	0.8
Infrastructure	0.7
Objects	0.6

Table 6: General Classes and Weights for affected objects

Then each class weight is multiplied with the number of the objects of that class, for example if the affected objects are 2 cars, then: *Total_Weight = 2 * Vehicles_Weight.* The Sum of the total weight of every item listed is then categorized as “Low”, “Medium”, “High” or “Very High” severity using predefined ranges (Table 7) and a message is sent to the KB with the total Severity Level, the geolocation and the id of the current POI.

Severity Level	Range
Low	[0.0, 25.0)
Medium	[25.0, 50.0)
High	[50.0, 75.0)
Very High	[75.0, 100.0]

Table 7: Severity Level and its range according to the total weight

For the Visual Analysis messages a Machine Learning (ML) approach was developed in order to assess the weights of every category that was detected from the Visual module with an intelligent way, and thus avoiding the use of heuristic weight values. Using the results produced by the analysis of example frames of videos with flood scenarios by the Visual Analysis module we created an annotated dataset with 96 entries. That dataset was then used to train and test four well-known ML algorithms, namely the Ridge Classifier, the Support Vector Machines (SVM), the Random Forest (RF) Classifier and the Decision Tree (DT) Classifier looking for the best possible accuracy score Then the best performing model was integrated into the DSS framework and by its interaction with the Knowledge Base, the updated values of the severity level of a POI can be inference (Figure 11).

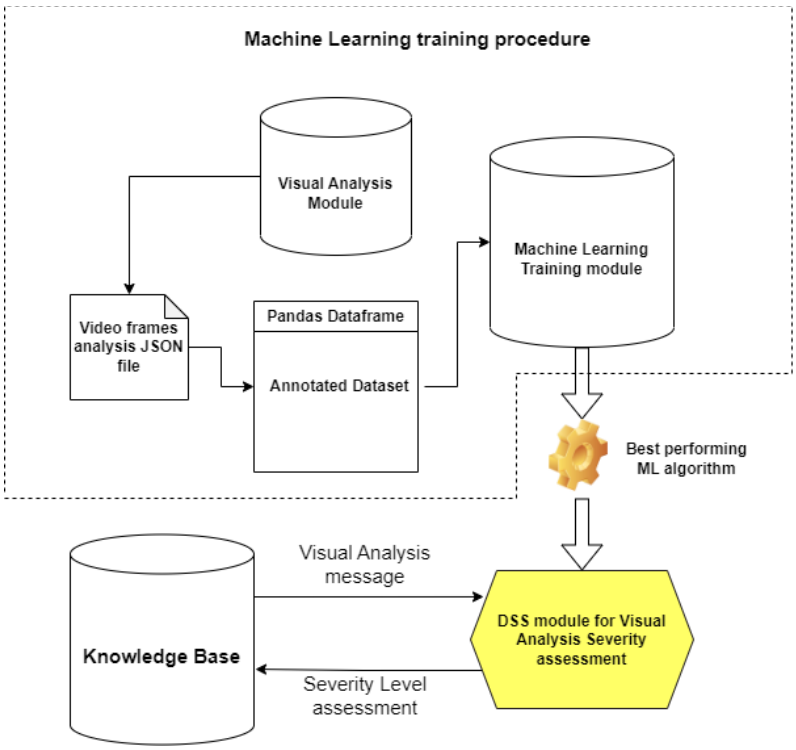

Figure 11: Machine Learning module development and KB interactions

Same as with the Textual Analysis, when a Visual Analysis message is generated KB sends the vital information DSS needs to assess the Severity Level of the event in real time. An example message can be seen bellow.

KB message to DSS for visual analysis severity assessment


```
{
  "dangerInfo":{
     "severity":"medium",
     "latitude":37.988,
     "created_at":"2022-11-17T13:34:16.898095+00:00",
     "id":"28",
     "longitude":23.75
  },
  "shotInfo":[
     {
        "area":"parking_lot",
         "riverOvertop":"",
        "objectsFound":[
           {
              "probability":0.5456057,
              "type":"water"
           }
        ],
       "outdoor":True,
       "objectsInDanger":[],
       "infraInDanger":["bridge"],
        "shotIdx":0,
        "peopleInDanger":1,
        "animalsInDanger":0,
        "emergencyProb":0.99999714,
        "emergencyType":"flood",
        "areaProb":0.1211,
        "vehiclesInDanger":5
     }
  ],
  "id":"28",
  "type":"Visual",
  "Timestamp":"2022.11.21.16.25.37"
}
```


Using the information contained in that message, DSS is able to use the pre-trained ML model in order to predict the Severity Level of the current situation. Then, DSS generates a message for the KB using the same format and the same SL classification categories as in the textual analysis (see 4.4.1 “DSS message for Creating/Updating Danger Zones”).

4.4 Danger Zone Creation or Update based on Severity Score

In this subsection we present how the danger zones are updated or created based on the severity score of the received from the DSS. Moreover, we will see how the severity score which is attached to the danger zones is fused with information coming from the stress level analysis component.

4.4.1 Danger Zone Creation and Update

As mentioned, danger zones are either created or updated from the KB, based on the severity score it receives from the DSS support system. There are three different scenarios, which we

will analyze in detail, that the KB considers when it receives a severity score from the DSS. The message received from the DSS in all scenarios has the following form.

DSS message for Creating/Updating Danger Zones

```
{
 "severityDSS": "medium",
 "latitude": "37.988",
 "longitude": "23.75",
 "project_id": 28
}
```

The label severityDSS can take four discrete values which indicate the severity level of a location (“Low”, “Medium”, “High”, and “Very High”). The latitude and longitude indicate the location that the severity score refers to, and the project_id is a unique identifier for the broader area that the location in the messages belong into.

- In the first scenario no danger zones exist in the area surrounding the point (i.e., latitude-longitude) that is contained in the message. In this case a new danger zone will be created that will be assigned the severity which is contained in the message. The danger zone is a bounding box which is created from the latitude and longitude that exist in the DSS message, based on the following function: (latitude-0.001 longitude - 0.001 latitude + 0.001 longitude + 0.001).
- In the second scenario, a danger zone already exists in the area. If the severity score is “Low” then we do not make any changes in the old danger zone.
- In the third scenario, a danger zone already exists in the area. If the severity score is “Medium”, “High”, or “Very High” then we apply some changes to the old danger zone following the next rules. If the new danger zone is totally contained in the old danger zone, then the area remains as it is, but if the new danger zone is not totally contained in the old danger zone, we will merge the area of the two danger zones (see Figure 12).

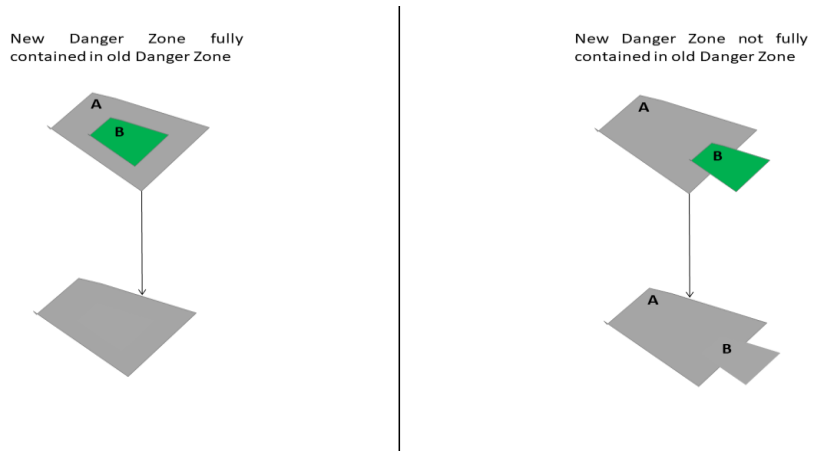

Figure 12: Creation or Update of Danger Zones

The endpoint for the danger zone creation/update based on the messages from the DSS is: *http://xr4drama.iti.gr:8090/xr4d_Retrieval/webapi/dangerZoneSeverity*.

4.4.2 Danger Zones based on Severity of Stress Level Analysis and Visual analysis fusion

The fusion of stress levels of first responders with results from visual analysis module can offer a better insight into the situation of the severity of the danger zones in real time. The fusion method is based on a fuzzy inference system which combines the stress level of each first responder with results from the visual analysis of the same danger zone in the last one hour.

The fuzzy inference system is a system based on fuzzy rules for the combination of different inputs. At first a set of fuzzy rules for the fusion of different inputs must be determined. Then the input values must be fuzzified using a membership function. After the inputs are fuzzified, they are combined according to the fuzzy rules. From the combination the rule strength is used as the fuzzified output. The outcome must then be defuzzyfied using again a membership function.

In the current application, the inputs are the stress levels of the first responders and the emergency probability from the visual analysis tool, and the output is the severity score of the current danger zone. All of these variables are fuzzified using a Gaussian membership function. Stress levels and severity scores are categorized into four categories (low, medium, high, very high) and emergency probabilities into two categories (low, high) and the fuzzy rules are presented in the Table 8.

Stress levels	Emergency probability	Severity score
low	low	low
medium	low	low
high	low	medium
very high	low	high
low	high	medium
medium	high	medium
high	high	high
very high	high	very high

Table 8: Fuzzy rules for the fusion of stress levels and emergency probabilities from visual analysis tool for the computation of severity score

For each stress level, all of the emergency probabilities from the visual analysis tool referring to the same danger zone in the last one hour are aggregated before getting fed into the fuzzy inference system.

4.5 Task Creation List from Decision Support System

The task creation list is another function which is a result of the communication between the KB and DSS. In this case, the DSS will recommend to the KB to attach a set of tasks to POIs, that need to be performed by the first responders for the disaster to be tackled. The use case for this service was based on a flooding scenario, in which external sensors would measure the water level of the river Bacchiglione which passes through Vicenza, and if the water level would overcome a threshold, then the DSS would recommend various tasks that need to be performed on different locations. Therefore, the task would be attached to POIs for the first responders to see and perform them, to have the best possible outcome in a disaster management scenario.

We provide a message as it comes from the DSS, which the KB must map in the various POIs existing on the map. Notice that where the task will be mapped is subject to the coordinates it contains.

Task List Creation Message


```
{
  "data": [
    {
      "case": 3,
      "dataora": "2022-08-27T13:00:00",
      "idrunhec":2351,
       "stationid":"33c",
      "name":"Sand packs distribution points (part 3/4)",
      "responsibility":"AIM Valore Città",
      "criticity":"Sand packs distribution points indicated in the
                  Municipal Civil Protection plan",
      "action":"Prepare Sandpacks distribution point (part 3)- Street
                Natale Del Grande 140 sand sacks",
      "threshold":"Avverse condizioni meteo",
      "waterlevel":30,
      "latitude":35.497,
      "longitude":109.144
    },
    {
      "case":3,
      "dataora":"2022-08-27T13:00:00",
      "idrunhec":2351,
      "stationid":"33d",
      "name":"Sand packs distribution points (part 4/4)",
      "responsibility":"AIM Valore Città",
      "criticity":"Sand packs distribution points indicated in the
                   Municipal Civil Protection plan",
      "action":"Prepare Sandpacks distribution point (part 4)- San
                Pietro Intrigogna 140 sand sacks",
      "threshold":"Avverse condizioni meteo",
      "waterlevel":30,
      "latitude":35.497,
      "longitude":109.144
    }
  ]
}
```


The most crucial information in the previous message is contained in the labels: (i) **name** which is the title of the task that needs to be performed, (ii) the **responsibility** which indicates who oversees performing the task, (iii) the **criticity** which show the volume of how important is for the task to be performed, and (iv) the **action** which contains a detailed description of the task. Apart from the important information, the water level, the coordinates, and the threshold for the water level are contained in the message. The endpoint for the task list creation based on the messages from the DSS is: *http://xr4drama.iti.gr:8090/xr4d_Retrieval/webapi/dss*.

4.6 General Information-Emergency Data-Legal Documents

This section refers to the crucial information that accompanies a project. A project is a set of observations, where each observation corresponds to visual, textual or stress level analysis

message. More intuitively, a project is a set of observation which are clustered based on locality. For instance, a project may refer to all the observation for the urban area of Vicenza.

- The **general information** refers to a general description of the area that the project refers to. In more detail, each time a new project is created the KB will extract from Wikipedia initially, and if it does not find any information from Wikipedia it will search in DBpedia, the abstract that refers to the location of the project.
- The **emergency data** are the emergency numbers that can be used for each project, the emergency numbers are subject to the country, and sometimes region of the country. These numbers are the numbers from which one can call the police, ambulances, and firefighters.
- The **legal documents** refer to the legislation concerning recording in public places for each country (and region of country in some cases), and legislation about operating drones in public spaces.

A detailed message of what this information looks like to the user is given below, where we extract all of the previous information for the capital of Germany, Berlin.

Information Retrieval Output


```
{

  "country": "Germany",
  "location":"Berlin",
  "emergency number": {
     "police": "110",
      "firefighter": "112",
      "ambulance": "112"
   }
  "description": "
Berlin is the capital and largest city of Germany by both area and popul
ation. Its 3.7 million inhabitants make it the European Union's most pop
ulous city, according to population within city limits. One of Germany's
sixteen constituent states, Berlin is surrounded by the State of Branden
burg and contiguous with Potsdam, Brandenburg's capital. Berlin's urbana
rea, …",
  "legislationDrone": "Unbemannte Luftfahrt
Informationen für Geflüchtete aus der Ukraine und ehrenamtlich Helfende
/ Інформація для біженців з України і для волонтерів: berlin.de/ukraine.
Informationen zum Coronavirus: berlin.de/corona Begriffsbestimmungen
UA – Unmanned Aircraft – Unbemanntes Luftfahrzeug:
Bezeichnet ein Luftfahrzeug, das ohne einen an Bord befindlichen Piloten
 autono oder ferngesteuert betrieben wird oder dafür konstruiert ist.
UAS…",
  "legislationVideo":
"Sie planen Dreharbeiten in Berlin? Hier finden Sie erste wichtige Infor
mationen. Ob Videoclips, Spielfilme, TV-
Serien oder Magazine – die Hauptstadt ist eine beliebte Filmkulisse.
Bereits mit kleinen journalistischen Film- und Fototeams, die Foto- oder
 Filmarbeiten auf öffentlichem Straßenland und Plätzen in Berlin und…",
 "uri": "https://en.wikipedia.org/wiki/Berlin"}
```

5 SEMANTIC REASONING FOR DECISION SUPPORT

A high-level reasoning architecture is illustrated in Figure 13. Briefly, we can say that the reasoning framework extends the xR4DRAMA semantic models to predefined rules that formulated based on the available context (e.g., metadata collected from the analysis results, population of the KB). The semantics are used to acquire an early understanding of the available contents and dependencies among the multimodal results in the form of interlinked data. The knowledge graphs that formed are used as an input to the reasoning tool that triggers the necessary reasoning rules to export additional knowledge. For a better understanding, the reasoning framework can be seen as a schema that combines data integration and interpretation.

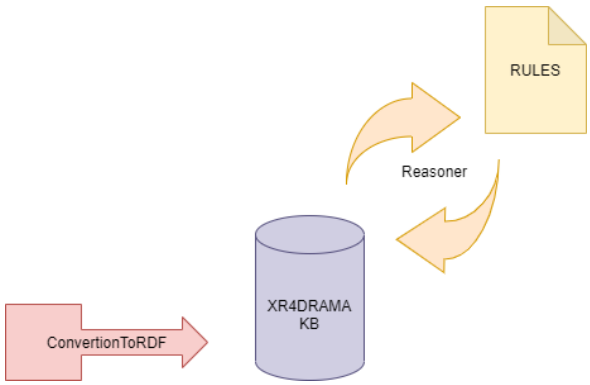

Figure 13: High level architecture of Semantic Reasoning

Apart from semantically analysing and correlating metadata, the reasoning framework excels at providing more complex searching capabilities to the end users, elaborating the SPARQL mechanics. This module is still in progress, and it will be further refined and presented in upcoming deliverable. The subsections that follow present some tasks that will be handled by the reasoning framework. The reasoning module is under development, and it will be refined and presented via a later WP3 deliverable, with a more extensive view of the data fusion and clustering. In the following subsections, some basic form of the tasks that can be handled by the reasoning framework, and an introduction to the statistical clustering.

5.1 Discovering Semantic Relations

There is the need to discover links among the heterogeneous data stored in the knowledge base, like audio, video, and social media content. Multimodal clustering on the different types of data will reveal common semantic information among them. For this task to be achieved, two approaches are being considered. First, the usual statistical clustering approach, which is a type of unsupervised learning. To apply statistical clustering on a set of different modalities, the latter need to be aligned, e.g., both audio and video recordings were recorded at the same timestamps. Secondly, semantic clustering approach that will be applied on social media

content to discover semantic relations among the different types of information that can be found in e.g., a tweet.

Clustering is a type of unsupervised classification method. This type of analysis finds common characteristics in a set of observations and assigns the observations to groups according to these common traits (Diday et al., 1976). The most popular clustering methods are K-means and hierarchical clustering. K-means groups observations into k clusters. Each cluster comprises of observations with means near the cluster centre. This method reduces the within clusters variances (Hartigan et al., 1979). Hierarchical clustering is a method that creates clusters with the top-down or bottom-up approach. In the first approach all cases belong to one cluster and as the recursive algorithm proceeds, they are divided into more (divisive approach) and in the second approach each case forms an individual cluster and as the algorithm proceeds, clusters are merged (agglomerative approach) (Maimon et al., 2005). Both clustering techniques require numeric variables.

For mixed types of variables, i.e., numeric and categorical, other types of clustering are available. K-prototypes is an extension of k-means for clustering numeric and categorical data (Huang 1998). K-prototypes combines k-modes, which is the k-means version for categorical data with k-means for clustering numeric data. k-modes creates clusters according to the matching categories of the variables.

In xR4DRAMA, the data stored in KB involve both categorical and numeric results, thus an approach like k-prototypes will be followed. However, it seems more plausible to apply semantic clustering on the text data saved from web-crawling component, to retrieve and organize the multimodal information of social media data. Semantic clustering is used to group web content based on common vocabulary. This approach is found is situation awareness tasks (Kingston et al 2018). Semantic clustering approaches require some initial data cleaning, which involves removal of irrelevant text from the web retrieved content and then semantic trees are created from connected words, which later form the clusters.

5.2 Rules

We use SPARQL, to implement expressive reasoning rules, enabling property value propagation and instance generation. The core idea is to associate each reasoning task with one or more SPARQL rules that support specific reasoning functionality, e.g., aggregation of different Risk levels. In the following, we present examples of such reasoning and rules. More elaborate rule-based reasoning cases will be presented in detail in future versions of the framework and reported in upcoming deliverable.

The reasoning concepts that we aim to accomplish have as guidelines the following major themes:

- **Risks Aggregation:** The different components have analysis results that can lead to specific risk levels. The visuals from information regarding the entities in dangers and the emergency types. The stress level from the high levels of stressed that are recorded. To have a final Risk Report to send to the authoring tool, we will aggregate information from the metadata in order to produce it.
- **Citizens-To-Project:** The citizens that will be involved during the process of a project won't be able to know anything around it. With the location coordinates that will be

retrieved from their reports and the information that these reports will contain we will assume regarding the possible project they will belong so we can use this new knowledge in higher level of analysis.

- **Risk/Safe areas Alerts:** The reasoning engine will support initialization of safe or risk areas with their coordination and a possible description. During an emergency event we will process knowledge from the other components about the availability and the situation of each of the areas respectively and will inform the authoring tool, to proceed with the notification of the citizens and/or first responders.
- **Spatio-Temporal event related Information:** Certain analysis results of the other modules contain either temporal or geolocated information. The reasoning framework will group this kind of information so there can be an overall view of each disastrous event.

6 EVALUATION

The xR4DRAMA KG was evaluated with two separate methods. On the one hand, we analysed the xR4DRAMA KG's consistency and completeness using two separate evaluation techniques. By developing a set of competency questions that the KG must be able to answer with the knowledge it contains (subsection 6.1), we evaluated the comprehensiveness of the xR4DRAMA KG. Then, we evaluated the xR4DRAMA KG's consistency by seeing if it adhered to a particular set of SHACL constraints (subsection 6.1). On the other hand, by calculating the precision-recall-F1 scores utilized in information extraction systems (subsection 6.2), the POI management mechanism was evaluated. Finally, we evaluated the DSS in order to see if the task that it proposes and the severity scores that it sends to the KB, to attach to POIs, are valid and accurate (subsection 6.3).

6.1 Completeness and Consistency of the Knowledge Graph

Competency Questions (CQs) compiled during the creation of the official ontology requirements specification document (ORSD) were used to assess the completeness of the xR4DRAMA KG (Suarez- Figueroa et al., 2009). For this reason, we asked a group of specialists to create a series of questions that they would like the xR4DRAMA KG to answer before we built it. The experts were authority workers from *Autorita' di bacino distrettuale delle alpi orientali* and journalists from *Deutsche Welle*. A total of 32 CQs were gathered, and we have included a sample of 10 of them in Figure 14. The full list of CQs may be accessed here.

1) What is the risk level of the observation?
2) Which is the emergency in the observation?
3) What is the detection/creation time of the observation?
4) Which is the area in the observation?
5) What is the probability of the area in observation?
6) Which is the Stress level of the between time intervals?
7) Which is the objects found in video?
8) Which is the most/least risky observation?
9) What is the multimedia type used in observation?
10) How many people are in danger between time intervals $[t1]$-$[t2]$?

Figure 14: Batch of Competency Questions 1

The completeness of the xR4DRAMA KG was found adequate, as each CQ when translated into a SPARQL counterpart returned the desired information. For instance, the fourth CQ from Figure 14 when translated into a SPARQL counterpart (see Example 1), for the observation VisualMetadata_2c60537511c240c9add7fb2eb4e7459e_0 returned amphitheater. If the observation is visual, the name will be created from the text VisualMetadata (if not, TextualMetadata) and a unique simmoid value.

Example 1.

```
SELECT DISTINCT ?area WHERE
{
  xr:VisualMetadata_2c60537511c240c9add7fb2eb4e7459e_0
  xr:hasInformationOfInterest ?info .
  ?info xr:hasLocation ?location .
  ?location xr:hasArea ?area.
}
```

In addition to the CQs, we carried out a validation process to examine the syntactic and structural quality of the KB's metadata and to verify their consistency. Custom SHACL consistency checking rules and native ontology consistency checking, such OWL DL reasoning, were used to adhere to the closed-world criterion. One can find constraint violations, such as cardinality inconsistencies, incomplete, or missing information, by employing the first method. By employing the latter, the terminological semantics, or TBox, are taken into account as validation, much like in the case of class disjointness. Out of 56 SHACL rules, 21 of which referenced to object type properties and 35 to data type properties, the consistency of the xR4DRAMA KG was deemed sufficient because none of them returned any rule invalidation. We also looked for instances that belong to the intersection of classes because we did not want that to happen, but none were found.

6.2 Information Retrieval System Validation

The standard precision, recall, and F1-score used for information extraction systems (Equations 1, 2 and 3) were applied to the evaluation of the POI management mechanism in order to create or update POIs from visual and textual messages. The POI management mechanism can be regarded as an information extraction mechanism, as a query is received (in our case a message from the textual or visual analysis components), and some information is extracted from the KG (a POI is created or updated).

$$precision = \frac{|\{\text{RelevantInstance}\} \cap \{\text{RetrievedInstance}\}|}{|\{\text{RetrievedInstance}\}|} \qquad (1)$$

$$recall = \frac{|\{\text{RelevantInstance}\} \cap \{\text{RetrievedInstance}\}|}{|\{\text{RelevantInstance}\}|} \qquad (2)$$

$$F1 = 2 * \frac{\text{recall} * \text{precision}}{\text{recall} + \text{precision}} \qquad (2)$$

Retrieved Instances are considered all the visual (or textual) messages for which the POI management mechanism, did not return an error when we casted a message in order to create or update a POI.

Relevant Instances are considered all the the visual (or textual) messages for which the POI management mechanism, managed to create or update a POI, when we casted a message with them.

The intuition behind the retrieved and relevant instances, is that retrieved pairs from the moment that the POI management mechanism did not return any error they can retrieve information (through a POI), while relevant are instance which managed to create or update a POI and therefore contain information relevant to a project.

The number of retrieved textual and visual messages are indicated by the variables $Retrieved_t$ and $Retrieved_v$, respectively. The numbers of relevant textual and visual messages are $Relevant_t$ and $Relevant_v$, respectively. Next, $recall_t$, $recall_v$, are the recall scores for the textual and visual messages, $F1_t$, $F1_V$ are the F1 scores for the textual and visual messages, respectively, and $precision_t$, $precision_v$ are the precision scores for the textual and visual messages.

One can find the dataset used to test our POI management mechanism here. It consists of a set of 1501 text messages and 800 visual messages. Be aware that the values of each label in each message were chosen at random from a gold standard dataset assembled by domain experts, in order to tackle potential biases. It is interesting that all messages—textual or visual—were considered to have been successfully retrieved, which means that our POI management mechanism never returned an error for any given message, whether it was textual or visual. The resulting values are $Retrieved_t$= 1501 and $Retrieved_v$ = 800. The same does not hold for the relevant messages, either textual or visual, as there were 1376 $Relevant_t$ messages for the textual analysis component and 697 $Relevant_v$ messages for the visual analysis component.

Table 9 contains the precision, recall, and F1 scores for both textual and visual messages based on the data. The results are rounded to four decimals.

	Precision	Recall	F1
Textual Messages	0.917	1.0	0.956
Visual Messages	0.871	1.0	0.931

Table 9: Precision, Recall and F1-scores for textual and visual messages

6.3 Decision Support Validation

Since the Textual Analysis Severity assessment is done by a heuristic rule-based approach, in order to validate the output of DSS we used the information contained in the annotated dataset from the analysed video frames by the Video Analysis module. After extracting the information from the flooded pictures and considering the input fields of the Textual Analysis messages, we modified the dataset in order to represent the same information every visual scene carried but in a textual message format. That information then was processed by the Textual Analysis DSS module, and the Severity Level generated was compared to the original severity value of the annotated visual dataset.

$$Accuracy\ score = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \qquad (1)$$

Textual Analysis DSS module was able to achieve a 0.766 accuracy score (1) using a total of 30 entries for testing.

In order to validate the Visual Analysis message severity assessment of DSS standard metrics like precision, recall, F1-score and Accuracy score were used (see 6.2 for definition). The annotated dataset was split at 76 entries for training and 20 entries for testing using cross validation. The four different Machine learning algorithms were tested after hyperparameter tunning each one. In the following table we can see the results for each one of the models according to their best achieved accuracy score.

	Accuracy Score
Linear Regression (Ridge Classifier)	0.65
SVM (SVC)	0.75
Random Forest Classifier	0.75
Decision Tree Classifier	0.85

Table 10: Machine Learning algorithms performance according to Accuracy score

The best performing algorithm is Decision Tree Classifier with max_depth = 5 (the maximum depth of the tree), max_leaf_nodes = 10 (the maximum amount of nodes that can be used when growing a tree in best-first fashion) and random_state = 42 (controls the randomness of the estimator – if integer a deterministic behaviour during fitting is obtained). In the following table we can see more analytically the validation metrics for DT.

	precision	recall	f1-score	support
Low	1	0.666667	0.8	3
Medium	0.75	1	0.857143	3
High	0.875	0.875	0.875	8
Very High	0.833333	0.833333	0.833333	6
accuracy	0.85	0.85	0.85	0.85
macro avg	0.864583	0.84375	0.841369	20
weighted avg	0.8625	0.85	0.848571	20

Table 11: Precision, Recall, F1-score for DT algorithm

Following there is the confusion matrix of the testing phase of DT algorithm, with the best performing parameters.

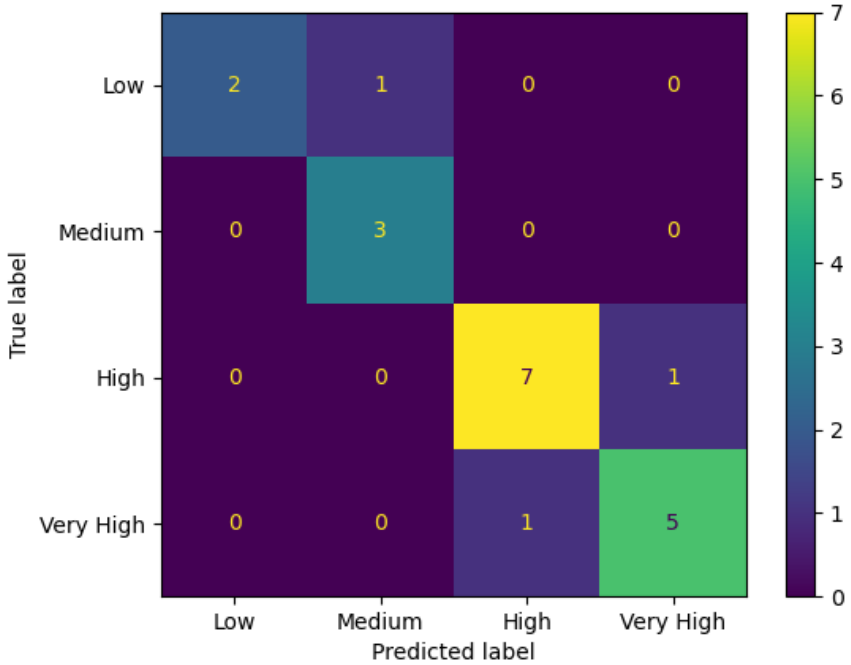

Figure 15: Confusion Matrix of DT algorithm

7 CONCLUSIONS AND FUTURE WORK

The xR4DRAMA KG was created, to serve as the knowledge representation for the xR4DRAMA project. Using many technologies, including eXtended Reality (XR), xR4DRAMA is committed to enhancing situation awareness. Media planning is one of the xR4DRAMA project's primary use cases. In short, xR4DRAMA project stands in three key-points: (a) Facilitating the gathering of all necessary (digital) information for a particular, difficult or even dangerous situation that a media team faces, (b) Utilizing extended reality technologies to simulate an environment "as if on site" in order to accurately predict an event or incidence, and (c) Establishing a shared understanding of an environment and giving users of the project's platform (first responders in the field/control room, citizens, and journalists) the option to update representations of locations as events change, allowing them to comprehend and re-evaluate the effects of particular actions/decisions. As a result, the xR4DRAMA KG may incorporate the findings of several advanced analysis components that process multimodal data and depict the structures they produce (in this project, for the media use case, we integrate visual and textual analysis messages). Additionally, the xR4DRAMA KG provides an innovative method through its POI management mechanism that may generate or update POIs, which contain essential geospatial data that can make it easier for journalists to cover the production recordings.

As part of their day-to-day business journalists and other media houses produce news coverage in various locations. Despite thorough research and preparations, remote production planning very often runs into challenges and difficulties. Many depend on the characteristics of the individual location and the situation on the ground. These challenges can be circumstantial and organisational - like accessibility, noise, the presence of people, the lack of infrastructure (from electricity supply to parking space), the wrong choice of equipment or other filming restrictions. Hence, it is crucial for journalists and media houses to access information about the state of a location, such as the accessibility of the location, among others, in order to plan their media coverage.

Moreover, one of the main use cases of the xR4DRAMA project focuses on disaster management. Therefore, the xR4DRAMA KG can represent the structures and integrate the results coming from multiple advanced analysis components that process multimodal data (in this project we integrate visual, textual, and stress level analysis messages). Additionally, the xR4DRAMA KG through its POI management mechanism offers an innovative mechanism that can create or update POIs, which contain crucial geospatial information that is needed in a case of emergency.

The challenge is to combine all this information into a coherent image to give everyone a precise picture of the location and the situation on the ground in order to prepare themselves for a smooth and safe production. The xR4DRAMA KG can fill the gap in distribution of crucial knowledge to journalists, for them to be able to plan more efficiently the news coverage.

Regarding the evaluation our goal was to examine the POI management mechanism using the precisionrecall-F1 metrics for information extraction systems, as well as the completeness and consistency of the xR4DRAMA KG. CQs, which were gathered by experts, were used to assess the KG's completeness. More specifically, we converted each CQ into a SPARQL equivalent,

and we anticipated that each one would return results. This is evidence that our KG may deliver significant information in a broader media planning scenario. A series of 56 SHACL restrict rules, of which 21 related to object type properties and 35 to data type properties, were used to test the consistency of the KG; none of them resulted in the rule being invalidated. Additionally, we looked to see whether there were any instances that belonged to the intersection of the classes, but none were found. This demonstrates the coherence of our KG, proving that it is free of noise and contradicting information.

Our POI management mechanism received strong F1 scores for both the visual (93.1%) and textual (95.6%) message, demonstrating that it can be utilized independently to create/update POIs in a broad media planning scenario. Additionally, we can comment that this occurred when updating POIs, which means that new information could not be added to POIs that already existed in the area and were missing relevant instances for both textual and visual messages. The updated messages were straying outside of the bounding boxes of all existing POIs because each POI has a box around it. Be aware that the POIs' bounding boxes are part of a broader bounding box that encompasses the area that requires news coverage. It seems reasonable to consider a bounding box for the POIs and the area that requires news coverage; otherwise, we risk adding POIs to the area that are situated in a completely other area of the map.

The high Recall scores—100% for both the visual and textual messages—can also be mentioned. This essentially indicates that there were no textual or visual messages that indicated an error. If we examine the two scenarios in which an error may be returned, the reason for not doing so is clear: (i) The message's coordinates do not fall within a bounding box that designates the location where an event has occurred, or (ii) The user will identify a non-matching category-subcategory tuple. It is difficult for the user to choose the incorrect selection in both cases since the user sends messages using a mobile application (which is now private) that displays the permissible category-subcategory tuples, and the bounding box with a blue hue over an area.

In terms of future work, we intend to provide a method that will make the POIs more beneficial while making decisions. Additionally, we will provide POIs with a list of tasks that must be taken in order to complete a remote production mission more accurately.

8 REFERENCES

Andreadis, S., Moumtzidou, A., Apostolidis, K., Gkountakos, K., Galanopoulos, D., Michail, E., Gialampoukidis, I., Vrochidis, S., Mezaris, V., Kompatsiaris, I. 2020. "Verge in vbs 2020", In International Conference on Multimedia Modeling, p. 778–783. Springer.

Babitski G., Bergweiler S., Grebner O., Oberle D., Paulheim H., Probst F. 2011. "Soknos–using semantic technologies in disaster management software," in Extended Semantic Web Conference. Springer, p. 183–197.

Batziou, E., Ioannidis, K., Patras, I., Vrochidis, S., and Kompatsiaris, I. 2023. "Low-light image enhancement based on u-net and haar", In Proceedings of the 29th International Conference on Multimedia Modeling (MMM 2023). Springer.

Berven, A., Christensen, O. A., Moldeklev, S., Opdahl, A. L., Villanger, K. J. 2018. "News hunter: building and mining knowledge graphs for newsroom systems", NOKOBIT, (26), p.1–11.

Buttigieg P. L., Morrison N., Smith B. et al. 2013. "The environment ontology: contextualising biological and biomedical entities", Journal of Biomedical Semantics, vol 4(43). https://doi.org/10.1186/2041-1480-4-43

Chou, C.-H., Zahedi, F. M., Zhao, H. 2010. "Ontology for developing web sites for natural disaster management: methodology and implementation," IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, vol. 41 (1), p. 50–62.

Compton, M., Barnaghi, P., Bermudez, L., Garcia-Castro, R., Corcho, O., Cox, S., ... & Taylor, K. 2012. "The SSN ontology of the W3C semantic sensor network incubator group", Journal of Web Semantics, vol 17, p. 25-32.

Deborah L., McGuinness, F. van H. 2004. "Owl web ontology language overview", W3C recommendation 10.2004-03.

Diday, E., Simon, J. C. 1976. "Clustering analysis", In Digital pattern recognition, p. 47-94. Springer.

Hartigan, J. A., Wong, M. A. 1979. "Algorithm AS 136: A k-means clustering algorithm", Journal of the royal statistical society. series c (applied statistics), 28(1), p. 100-108.

Hobbs J. R., Pan F. 2004. "An Ontology of Time for the Semantic Web", CM Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing, vol 3 (1), p. 66-85.

Hoill J., Chung K. 2015. Ontology-driven slope modeling for disaster management service. Cluster Computing, vol 18, p. 677–692. https://doi.org/10.1007/s10586-015-0424-1

Huang, Z. 1998. "Extensions to the k-means algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery vol 2(3), p. 283-304.

Janowicz, K., Haller, A., Cox.J.D., S., Le Phuoc,D., Lefrançois,M. 2019. "SOSA: A lightweight ontology for sensors, observations, samples, and actuators", Journal of Web Semantics, vol 56, p. 1-10. https://doi.org/10.1016/j.websem.2018.06.003

Kingston, C., Nurse, J. R., Agrafiotis, I., & Milich, A. B., 2018. "Using semantic clustering to support situation awareness on Twitter: the case of world views", Human-centric Computing and Information Sciences, vol 8(1), 1-31.

Klien E., M. Lutz, Kuhn W., 2006. "Ontology-based discovery of geographic information services—an application in disaster management," Computers, environment and urban systems, vol 30 (1), p. 102–123.

Knublauch H., Hendler J. A., Idehen K. (2011). SPIN - Overview and Motivation W3C: Member Submission 22 February 2011.

Li L., Li T., 2013. "An empirical study of ontology-based multi-document summarization in disaster management," IEEE transactions on systems, man, and cybernetics: systems, vol 44 (2), p. 162–171.

Maimon, O., Rokach, L. (Eds.) 2005. "Data mining and knowledge discovery handbook. Springer.

Masmoudi, M., Karray, M. H., Lamine, S. B. A. B., Zghal, H. B., & Archimede, B. 2020. "MEMOn: modular environmental monitoring ontology to link heterogeneous earth observed data", Environmental Modelling & Software, vol 124, 104581, https://doi.org/10.1016/j.envsoft.2019.104581

Mazimwe A., I. Hammouda, Gidudu A., "Implementation of fair principles for ontologies in the disaster domain: A systematic literature review," ISPRS International Journal of Geo-Information, vol 10 (5), p. 324, 2021.

Moreira, J. L., Ferreira Pires, L., van Sinderen, M., Costa, P. D. 2015. "Towards ontology-driven situation-aware disaster management", Applied ontology, 10(3-4), 339-353.

Murgante, B., Scardaccione, G., Casas, G. L. 2009. "Building ontologies for disaster management: seismic risk domain", In Urban and Regional Data Management, p. 271-280. CRC Press.

Noy, N. F., & McGuinness, D. L. 2001. "Ontology development 101: A guide to creating your first ontology".

Opdahl, A. L., Berven, A., Alipour, K., Christensen, O. A., Villanger, K. J. 2016. "Knowledge graphs for newsroom systems", NOKOBIT, vol 24, p. 1-4.

Pan F., Hobbs J. R. 2004. "Time in OWL-S", Proceedings of the AAAI Spring Symposium on Semantic Web Services, p. 29-36. Stanford University.

Pan, J. Z., Pavlova, S., Li, C., Li, N., Li, Y., and Liu, J. (2018). Content based fake news detection using knowledge graphs. In International semantic web conference, pages 669–683. Springer.

Pérez, J., Arenas, M., & Gutierrez, C. 2009. "Semantics and complexity of SPARQL", ACM Transactions on Database Systems (TODS), 34(3), p. 1-45. https://doi.org/10.1145/1567274.1567278

Purohit, H., Kanagasabai, R., Deshpande, N. 2019. "Towards next generation knowledge graphs for disaster management", In 2019 IEEE 13th international conference on semantic computing (ICSC), p. 474-477. IEEE.

Qiu, W., Li, W., Liu, X., & Huang, X. 2021. "Subjective street scene perceptions for shanghai with large-scale application of computer vision and machine learning", Technical report, EasyChair.

Rospocher, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T., Bogaard, T. 2016. "Building event-centric knowledge graphs from news", Journal of Web Semantics, vol 37, p.132–151.

S. Werder, 2007. "Knowledge representation for disaster management," in Proc. of International Symposium on Strong Vrancea Earthquakes and Risk Mitigation, p. 4–6.

Studer, R., Benjamins, V. R., Fensel, D. 1998. "Knowledge engineering: Principles and methods", Data & knowledge engineering, 25(1-2), 161-197. https://doi.org/10.1016/S0169-023X(97)00056-6

Wang, Y., & Hou, X. 2018. "A Method For Constructing Knowledge Graph Of Disaster News Based On Address Tree", In 2018 5th International Conference on Systems and Informatics (ICSAI), p. 305-310. IEEE.

Wise C., Ioannidis V. N., Calvo M. R., Song X., Price G., Kulkarni N., Brand R., Bhatia P., Karypis G., 2020. "Covid-19 knowledge graph: accelerating information retrieval and discovery for scientific literature," arXiv preprint arXiv:2007.12731.

Xu, W., Zlatanova, S. 2007. "Ontologies for disaster management response", Geomatics Solutions for Disaster Management, p. 185-200.

Zhang Y., Zhu J., Zhu Q., Xie Y., Li W., Fu L., Zhang J., Tan J., 2020. "The construction of personalized virtual landslide disaster environments based on knowledge graphs and deep neural networks," International Journal of Digital Earth, vol. 13 (12), p. 1637–1655.