big data and cognitive computing

MDPI

Article

GTDOnto: An Ontology for Organizing and Modeling Knowledge about Global Terrorism

Reem Qadan Al-Fayez [1,*], Marwan Al-Tawil [1], Bilal Abu-Salih [2] and Zaid Eyadat [3]

[1] Computer Information Systems Department, King Abdullah II School of Information Technology, The University of Jordan, Amman 11942, Jordan
[2] Computer Science Department, King Abdullah II School of Information Technology, The University of Jordan, Amman 11942, Jordan
[3] Prince Al Hussein bin Abdullah II School of International Studies, The University of Jordan, Amman 11942, Jordan
* Correspondence: r.alfayez@ju.edu.jo

Abstract: In recent years and with the advancement of semantic technologies, shared and published online data have become necessary to improve research and development in all fields. While many datasets are publicly available in social and economic domains, most lack standardization. Unlike the medical field, where terms and concepts are well defined using controlled vocabulary and ontologies, social datasets are not. Experts such as the National Consortium for the Study of Terrorism and Responses to Terrorism (START) collect data on global incidents and publish them in the Global Terrorism Database (GTD). Thus, the data are deficient in the technical modeling of its metadata. In this paper, we proposed GTD ontology (GTDOnto) to organize and model knowledge about global incidents, targets, perpetrators, weapons, and other related information. Based on the NeOn methodology, the goal is to build on the effort of START and present controlled vocabularies in a machine-readable format that is interoperable and can be reused to describe potential incidents in the future. The GTDOnto was implemented with the Web Ontology Language (OWL) using the Protégé editor and evaluated by answering competency questions, domain experts' opinions, and running examples of GTDOnto for representing actual incidents. The GTDOnto can further be used to leverage the publishing of GTD as a knowledge graph that visualizes related incidents and build further applications to enrich its content.

Keywords: ontology; semantic web; social data; terrorism; OWL/RDF; knowledge graphs

check for updates

Citation: Al-Fayez, R.Q.; Al-Tawil, M.; Abu-Salih, B.; Eyadat, Z. GTDOnto: An Ontology for Organizing and Modeling Knowledge about Global Terrorism. *Big Data Cogn. Comput.* **2023**, *7*, 24. https://doi.org/10.3390/bdcc7010024

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 19 December 2022
Revised: 18 January 2023
Accepted: 19 January 2023
Published: 28 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Recent studies have emphasized the importance of publishing open data despite the challenges faced [1,2]. Having datasets available and stored in data portals or repositories online offers more value for the data, especially when such datasets are available for researchers, businesses, and government to utilize [3]. Researchers and practitioners in different institutions have endorsed publishing datasets as open data [4]. The integration of several open datasets helps in making better decisions in general. Semantic web technologies allow the publishing and sharing of data in machine-readable formats that ease data integration and enable knowledge sharing and analytics capabilities [5–8]. The best example is the recent COVID-19 pandemic, in which sharing open datasets about the viruses and experimental datasets from previous literature enabled scientists to develop vaccines to save humanity in record time [9].

Finding trustworthy datasets is challenging, especially with the abundant datasets available and published under licensing conditions, in different formats, and with varying metadata standards [10]. Technical challenges for publishing open datasets, such as data replication and lack of standards for describing metadata, are discussed in the literature [11]. Furthermore, the semantic web revolutionized the publishing of information on the web

since semantic technologies, such as ontologies and the RDF data model, replaced other common and widely used data models, such as HTML, XML, JSON, spreadsheet, or text files [12]. Such techniques solve data ambiguity, interoperability, and integration issues when published online.

The scarcity of publicly available semantic datasets published in social science is the motive behind this study. Literature about semantic web portals in more specific fields, such as world terrorism, is limited. To our knowledge, a few studies have worked in this domain. Some terrorist incidents are chains of operations reported by a small group of people in different places. The semantic modeling of such incidents will facilitate checking terror ties between operations after representing terrorist incidents. In [13], researchers realized the potential utilizing ontologies in analyzing terrorist network. When investigating terrorism, valuable information and external information about incidents, people, and targets are immensely needed. It is hard to label any violent attack as a terrorist incident, given the saying, "Terrorism is in the eyes of the beholder". The semantic modeling of social data, including terrorism information can serve as a valuable tool for terrorism investigators. It might not succeed in identifying links between operations happening in real time. Still, it can gather and organize information and help investigators to better explore and link information about a specific situation. For instance, disambiguating attacker names or places mentioned in terrorism incidents over media and social media can be performed via the semantic web. Referring to characters or places using URIs will enable better exploration of related information without hard linking articles and news. Hence, the semantic web will facilitate the intuitive process of exploring information and seeing patterns.

In the field of terrorism data organization, non-computer science specialists built an ontology for terrorism analysis and published it in an operational semantic web portal, Profiles in Terror (PiT) [14]. The researchers confessed that developing an ontology covering all aspects of terrorist activities is time-consuming. The semantic web portal link provided in the project (profilesinterror.mindswap.org) was not available at the time of writing this study. Another research found in the literature developed an ontology to organize the data collected from news articles about terrorism [15].

A team of multidisciplinary researchers at the University of Maryland worked on developing the criteria and attributes of each potential terrorist incident. With more than 50 years of experience, the National Consortium for the Study of Terrorism and Responses to Terrorism (START) at the University of Maryland maintained a project named the Global Terrorism Database (GTD) [16]. The team at START has developed the best practices for collecting information about terrorist incidents from 1970 onwards, containing almost 200,000 records with more than 130 variables describing incidents around the world [17].

In this work, we build on the efforts of GTD and propose the GTD ontology (GTDOnto) to define the concepts, vocabularies, and relations to describe any terrorist incident in the GTD using the OWL format. Based on the NeOn methodology guidelines, we developed this ontology from scratch to define vocabulary and relations used for representing the incidents of the GTD dataset in a machine-readable and interoperable format. We evaluated the applicability of the proposed ontology by describing incidents of the GTD using GTDOnto, in addition to domain expert feedback and competency questions answering. The remainder of this paper starts with the background knowledge section, followed by the methodology section. Next, we discuss the results and evaluation of the proposed ontology. Finally, the paper concludes with a summary and future work plans for using the proposed ontology.

2. Background

2.1. Ontologies

Ontologies have been central in developing the semantic web [18]. Gruber provided a famous definition of ontology as explicit formal specifications of the terms in the domain and relations among them [19]. At its most basic, an ontology consists of classes, instances, and properties [7].

Classes: define the main concepts in the domain. An example of a class in the musical instrument domain is *Guitar*. Characteristics of classes apply to their instances or individuals. For example, the fact that *Guitar* has nick and strings can be used with instances (i.e., individuals) of *Guitar*, such as *Vietnamese Guitar*. Every ontology has a class hierarchy consisting of all classes linked via the subsumption relationship *rdfs:subClassOf*. The set of concepts in the class hierarchy can be divided into the following three types: (i) *root entity* (the superclass for all entities in the class hierarchy); (ii) *Category entities* (set of all inner entities other than the root entity, that have at least one subclass; and (iii) *Leaf entities* (set of entities that have no subclasses). Every subclass in the hierarchy inherits the characteristics of its superclass.

Instances: (also called individuals) can be concrete objects, such as people, animals, and musical instruments, or abstract individuals, such as numbers and words (strings). Every instance belongs to at least one class (i.e., one instance can belong to more than one class). An instance inherits the attributes of its class and has specific values that differentiate them from other individuals in the class.

Relationships: (also called properties) in an ontology are used to define the characteristics of the classes. Relationships have labels, and they represent links between classes and instances. For example, a necessary type of relationship between classes is the subsumption relationship *rdfs:subClassOf,* which is used to identify the subclass/ superclass relationships in the class hierarchy (also called the subsumption class hierarchy). While the subsumption relationship is typical in different ontologies, other relationship types called object properties and data properties relationships are domain-specific and used in ontology.

2.2. Ontology Development Methodologies

In the literature, various methodologies for developing well-founded domain ontology are summarized [20]. The Uschold and King Methodology is one of the first ontology development approaches proposed [21]. It includes four stages: the first stage defines the purpose of the ontology (i.e., why the ontology is being developed and its intended uses). The second stage focuses on building the ontology and consists of the following phases: capture the ontology by identifying key concepts and relationships, code the ontology in a formal language, and integrate existing ontologies. The third stage focuses on evaluating the ontology, and the final step is documenting the ontology.

The Human-Centered Ontology Engineering Methodology (HCOME) ontology development methodology has been proposed as an approach that considers the active participation of knowledge in the ontology life cycle [22]. The HCMOE has three phases for ontology development: specification, conceptualization, and exploitation. In the specification phase, knowledge workers collaborate in defining the scope and aim of the ontology and producing specification documents. The second phase acquires knowledge from existing ontologies to develop and maintain the ontology. The final phase focuses on utilizing and browsing the ontology within an application and evaluating the ontology.

The work of [23] proposed an iterative approach to a simple knowledge-engineering ontology development methodology. The approach consists of three steps that start with a rough ontology, then revise and refine the ontology and finally discuss the modeling decisions, including the pros and cons of these decisions. This iterative approach continues during the lifecycle of the ontology that starts with defining the domain and scope of the ontology. The domain represents the main concepts used in the ontology and uses competency questions to determine the scope of the ontology. These questions test whether the ontology has enough information to answer them. The second step is about reusing existing ontologies. The third step focuses on writing essential concepts. Then the fourth step uses top-down, bottom-up, or a combination of both to define the class hierarchy. In step five, classes' properties are defined, and in step six, slots (i.e., objects' properties) between classes are defined. Finally, in step seven, instances (i.e., individuals) of classes in the class hierarchy are created. Kanga methodology engages domain experts in ontology development [24]. This methodology combines two aspects: the conceptual aspect written

by domain experts. The logical aspect is performed by converting the conceptual knowledge into a machine-readable format, such as OWL. The methodology has five phases: (i) ontology requirements, where scope and purpose are identified; (ii) source knowledge capture, where knowledge sources and core concepts and relationships are identified; (iii) populating knowledge glossary, which covers the glossary of key concepts and relationships; (iv) formal structuring where a controlled natural language (OWL, RDF) is used to define concepts, relationships and axioms; and finally, (v) the evaluation and verification phase of the ontology using different techniques.

The NeOn methodology for ontology engineering suggests a variety of pathways for developing ontologies [25]. The methodology follows a Waterfall Ontology Network Life Cycle Model. This model describes four main phases for ontology development: (i) the initiation phase, which focuses on ontology requirements, (ii) the design phase, where the main concepts in the ontology are defined (i.e., ontology conceptualization), (iii) the implementation phase, which focuses on coding the logical ontology, and (iv) the maintenance phase, which concerns the documentation and validation of the developed ontology.

This study aims to develop a new ontology for organizing the description of global incidents based on the domain expert knowledge collected from the GTD project run by START. Hence, among the previously described ontology development methodologies, we selected the NeOn methodology since it provides scenarios for developing a new domain ontology and enables the development of an ontology from scratch using clear guidelines.

3. GTDOnto Development Methodology

This research adopts the NeOn methodology in developing the GTDOnto [26]. The NeOn methodology was followed in this research for several reasons: (1) It has been used in the literature to build ontologies in different areas by different people from various backgrounds [27–29]. (2) The NeOn methodology proposes several scenarios for developing an ontology, including a scenario for developing an ontology from scratch [30]. (3) Since this research aims to define controlled vocabulary for describing global terrorism-related incidents using RDF, the NeOn methodology enables having a glossary of terms that will be beneficial for building this vocabulary. (4) Compared to other methodologies for developing ontologies, NeOn is one of the most recent methodologies published in the literature and captures several older techniques presented in previous suggested work.

The proposed ontology in this work was developed by following the guidelines of Scenario 1: From Specification to Implementation. This scenario outlines the steps for implementing an ontology from scratch. Since there is little research about organizing information about terrorism, this scenario fits to develop GTDOnto. The proposed ontology will provide a standardized model to describe incidents from the GTD using entities and relations connecting these entities based on the GTD codebook.

The National Consortium START at the University of Maryland defined the variables used to describe incidents and published several research papers on their data collection methodology. Their work is documented in the GTD codebook and is used as the primary source for building the proposed GTDOnto [31]. Therefore, before starting with the NeOn methodology for developing GTDOnto, the acquisition process is explained in the next section.

3.1. Knowledge Acquisition

The current Global Terrorism Database (GTD) maintained by START is the product of several phases of data collection efforts. The data collected are published under a EULA agreement providing a conditional agreement to access and use the GTD. In addition, START offers an interface for browsing its content through its website (https://www.start.umd.edu/gtd/, accessed on 16 June 2021). As explained in their codebook, the data were collected based on media articles, electronic news archives, existing data sets, and other

sources, such as books, journals, and legal documents. Several parties performed the data-collection process over different periods, all documented in their codebook [31].

The efforts performed by START are immense. The START institute explained all the legacy issues in the data collected regarding the information available about which incidents based on the data-collection date. During the data collection process, they committed to coding some variables and were transparent in explaining the coding decisions wherever possible. Furthermore, their data-collection methodology added the inclusiveness criteria for each incident. Since the definitions of terrorism vary and START targets the public mass, they describe the inclusiveness criteria for considering an incident as an act of terrorism.

Additionally, START introduced the doubted variable to document if an incident was arguably doubted to be a terrorist incident. Based on the trustworthiness of the GTD data-collection process, the work of START is considered the source for developing GTDOnto in its first version. The attributes of a terrorism incident in the GTD are translated into classes and properties describing different entities related to an incident.

The downloadable version of GTD is available in a spreadsheet format. It consists of (191465) rows representing terrorist incidents and (135) columns to describe each incident. The codebook explains each attribute defined in the columns, its use, coding (if any), and other issues in the data collected for this attribute. The attributes explain details about each incident in several categories: (1) GTD ID, incident date, (2) incident location, (3) incident information, (4) attack information, (5) weapon information, (6) target/victim information, (7) perpetrator information, perpetrator statistics, claims of responsibility, (8) casualties and consequences information, and (9) additional information, and source information.

3.2. Specification Phase

For writing the ontology specification, NeOn provides precise guidelines to build the Ontology Requirement Specification Document (ORSD) [32]. The ORSD states why the ontology is being developed. Table 1 illustrates the process of developing GTDOno using the ORSD template.

Based on one of the NeOn methodology scenarios for developing an ontology, *Scenario 1: from specification to implementation*, it is recommended to use competency questions (CQs) to create the ontology requirements [30]. Therefore, in this first draft of GTDOnto ontology, we identified the need to answer multiple competency questions about several categories related to incidents in the functional requirements of the ORSD table. In addition, the ORSD identifies the key terms from the GTD codebook associated with a terrorist incident. A sample of these terms is summarized in Table 2 and will be further detailed to build the GTDOnto.

Table 1. GTDOnto ontology requirements specification document (GTDOnto ORSD).

Purpose
GTDOnto ontology stands for Global Terrorism Database Ontology. The GTDOnto ontology represents the knowledge necessary to describe an incident considered an act of terrorism as in the GTD.
Scope
GTDOnto ontology identifies entities representing attackers, targets or victims, perpetrators, weapons, and detailed information about the causalities and consequences. The ontology provides many attributes to define values for the following categories: 1. GTD ID, incident date, 2. incident location, 3. incident information, 4. attack information, 6. target/victim information, 7. perpetrator information, perpetrator statistics, claims of responsibility, 5. weapon information, 8. casualty information, consequences, kidnapping/hostage taking information, 9. additional information, and source information.
Implementation Language
GTDOnto ontology is implemented in OWL/RDF using Protégé. The implementation process is performed manually to define all top-level concepts of the ontology, and further automation is performed to build lower-level concepts and properties.

Table 1. *Cont.*

Intended End-Users
GTDOnto ontology is vital to (i) end users who may browse the dataset, (ii) curators interested in integrating datasets, and (iii) researchers who may use the data for network analysis and further analytics studies.
Intended Uses
GTDOnto ontology models the controlled vocabulary describing GTD incidents, such as attack types, weapon types, and target types. The ontology can be used to publish the incidents from the GTD to form a knowledge graph. In addition, the GTDOnto helps visualize the relations between potential attacks or attackers. Publishing the GTD using this ontology will result in a knowledge graph of related incidents that can be further explored for advanced data analysis. Furthermore, the possible knowledge graph representing incidents using GTDOnto can be enriched with content from other datasets and social media content.
Ontology Requirements
Non-Functional Requirements
Not applicable Functional Requirements
It can be set with groups of Competency Questions (CQG) that cover all the concepts in the incident. **CQG1: General information about an incident/incidents:** CQ1.1: How many incidents occurred in the year 2005? CQ1.2: What is the detailed location of the incident with ID ”200109110004”? CQ1.3: Is there any doubt that the incident with ID ”200109110004” is a terrorist incident or not? CQ1.4: If the incident is part of multiple events, what are the related incidents? . . . . . . **CQG2: Attack details related to an incident/incidents:** CQ2.1: What are the possible types of attacks associated with global incidents? CQ2.2: Was the terrorist attack successful, and was it a suicidal attack? CQ2.3: What are the attack types recorded in the incident with an ID ”200109110004”? CQ2.4: What are the incidents that recorded ”BombingExolosive” attacks? . . . . . . **CQG3: Weapons used in an incident/incident:** CQ3.1: What are the possible weapon types recorded with global incidents? CQ3.2: What are the possible weapons subtypes of ”explosives” weapons type? CQ3.3: What types and subtypes of weapons were used in the incident with an ID ”200109110004”? CQ3.4: What incidents used a weapon of type "explosives"? **. . . . . .** **CQG4: Targets and victims details related to an incident/incidents:** CQ4.1: What are the possible targets for global incidents? CQ4.2: What are the possible categories for a military-targeted attack? CQ4.3: What are the nationalities of all targets/victims of a terrorist incident? CQ4.4: Who is the specific target/victim of a terrorist incident? **. . . . . .** **CQG5: Perpetrators details for an incident/incidents:** CQ5.1: Does the terrorist incident claim responsibility by a group? If yes, what is the group name that carried out the incident? CQ5.2: How many individuals are reported to participate in the incident, and how many are taken into custody? CQ5.3: What methods are used to announce the claim of responsibility for a terrorist incident? **. . . . . .** **CQG6: Casualties and consequences of an incident:** CQ6.1: How many confirmed fatalities and injuries were reported in a terrorist incident? CQ6.2: How many perpetrators were killed and injured in a terrorist incident? CQ6.3: Is there any property damage reported in the incident? If yes, what is the damage to property that occurred in the incident? CQ6.4: Were there any hostages in the incident? What is the outcome of reported hostage/kidnapping incidents? What is the total number of hostages? **. . . . . .**

Table 2. Sample of terms found in GTD.

Incident	Attack	Target
Perpetrator	Weapon	Casualties
Ransom	Hijacking	Country/Nationality

3.3. Conceptualization Phase

Following the GTD codebook and the ORSD for the proposed GTDOnto, many concepts describe an incident illustrated in a conceptual model shown in Figure 1.

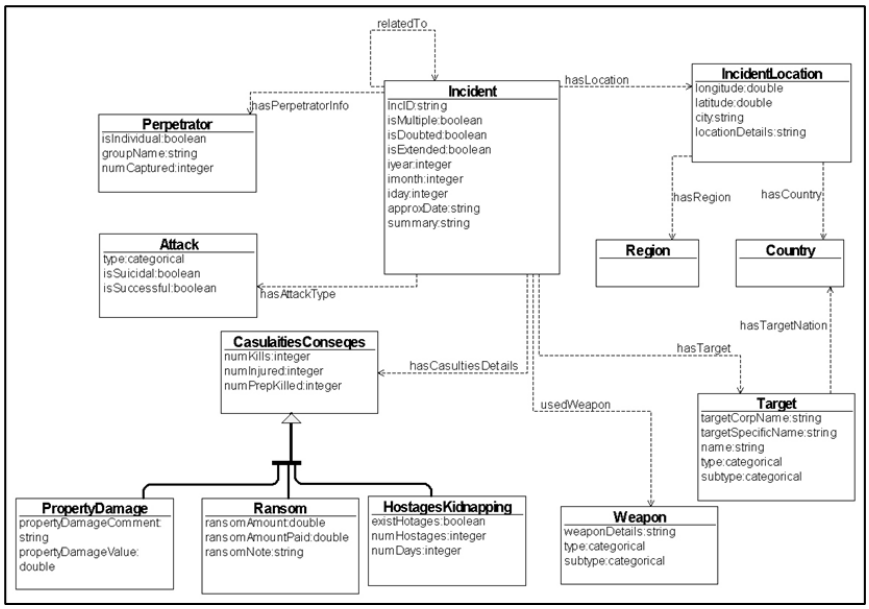

Figure 1. Conceptual model of GTDOnto ontology.

The conceptual model is structured in a class view, with subclasses showing further details related to each concept. The conceptual model defines the core concepts and examples of properties describing each concept. For example, the casualties and consequences class have further details, such as ransom demand, hijacking or kidnapping victims, or property damaged during an attack. In GTDOnto, the goal is to describe each class with properties and relations with other classes. Table 3 lists examples of the data properties needed to describe each class in a data dictionary.

Table 3. Data attributes examples.

Concept	Data Attributes	Description
Incident	id	A numeric variable follows a 12-digit Event ID system ex. "199307250001".
	day	A numeric variable records the day of the month on which the incident occurred.
	Month	A numeric variable records the month in which the incident occurred.
	year	A numeric variable records the year in which the incident occurred.
	approxdate	A text variable is used whenever the exact incident date is unknown or remains unclear. It records the approximate date of the incident.
	summary	A brief narrative summary of the incident, noting the "when, where, who, what, how, and why".

Table 3. *Cont.*

Concept	Data Attributes	Description
Attack	isSuccessful	A categorical type records whether the incident was successful or not. The definition of a successful attack depends on the type of attack. The key question is whether or not the attack type took place.
	isSuicidal	A categorical type records whether the incident was a suicide attack or not.
	type	A categorical type records the general attack method, consisting of nine categories.
Casualties and Consequences	numKills	A numeric variable stores the number of confirmed fatalities for the incident, including all victims and attackers who died as a direct result of the incident.
	numInjured	A numeric variable records the number of confirmed non-fatal injuries to both perpetrators and victims.
	numPerpKilled	A numeric variable limited to only perpetrator fatalities.
Hijacking	existHostage	A categorical variable records whether the victims were taken hostage or kidnapped during the incident.
	numHostages	A numeric variable records the total number of hostages or kidnapping victims.
	numHours	A numeric variable records the duration of the incident if the incident lasted for less than 24 h.
	numDays	A numeric variable records the duration of the incident in days if the kidnapping/hostage incident lasts more than 24 h.
	numReleased	A numeric variable records the number of hostages who survived the incident

After defining the data properties used to describe concepts in the GTDOnto, object properties are defined to describe the relations between different concepts. Table 4 represents a sample of possible relations between other concepts.

Table 4. GTDOnto object properties examples.

Object Property	Domain	Range
hasCountry	Incident	Country
hasAttackType	Incident	Attack
usedWeapon	Incident	Weapon
IsDoubted	Incident	DoubtStatus
hasAlternative	YesDoubted	AlternativeDesignation
hasHostKid	Incident	HostagesKidnappingStatus
hasHostKidOutcome	VictimsHostageKidnapped	HostKidnappingOutcome

3.4. Formalization Phase

After modeling the concepts, properties, and relations between concepts, the GTDOnto ontology is formalized. This process involves identifying subsumption relations and identifying domains and ranges for data and object properties. These properties are the semantic relations between pairs of classes to build relations between ontology instances in the future. The formalization phase identified classes, subsumption relations, object properties, and data properties. This version for the GTDOnto ontology is the first proposed version and is keen to further changes or addition to its properties after practice and usage of the ontology.

The subsumption relation builds the taxonomy of classes and subclasses in the GTDOnto ontology. For example, the *Weapon* class represents weapons used in the attacks, and it consists of 13 subclasses of weapons coded in the GTD codebook, such as *Biological*, *Chemical*, *Explosives*, *Firearms*, *Nuclear*, and others. Some of these weapon types have subtypes. For example, the *Chemical* weapon type can be *Explosive* or *Poisoning*. Meanwhile, the *Explosives* weapon type class has subtypes of weapons such as *Dynamite TNT*, *Grenade*, *Landmine*, and others.

The object properties build relations between classes. For example, to detail that some incidents in the GTD had hostages or kidnapping of victims, several features are recorded about this. In GTDOnto onology, the "*hasHostKid*" object property relates the *Incident* class with *HostagesKidnappingStatus* class that includes three subclasses to represent *NoVictimsHostageKidnapped*, *VictimsHostageKidnapped*, or *UnknownHostageKidnapped*. The GTD has cases where no information was recorded regarding hostages or the kidnapping of victims. Hence, the unknown status is required to confirm that information is missing in some incidents. If it was confirmed that victims were kidnapped or taken as hostages, further details are required to be recorded: *numHours*, *numDays*, *numHostKid*, *numReleased*, *KidhijCountry*, and others. Furthermore, the outcome of this attack is formalized by "*hasHostKidOutcome*" object property relating the Incident class with *HostKidnappingOutcome* class, which has seven possible outcomes represented as classes: *AttemptedRescue*, *HostagesEscaped*, *HostagesKilled*, *SuccessfulRescue*, and others.

To formalize the conceptual model and the relations for describing terrorism incidents, first-order-logic (FOL) was used before developing the GTDOnto. The FOL syntax defines knowledge about concepts in any domain as objects, relations, and functions close to natural human language. A sample of the FOL formulas and their representation statements are detailed below. These statements describe incidents and other concepts in the GTDOnto ontology:

- Any incident with victims taken as hostages or kidnapped has an attack-type of hijacking or hostages taken.

$$\forall x(\text{Incident(x)} \wedge \text{hasHostKid(x, } \textit{VictimsHostageKidnapped}) \rightarrow \text{hasAttackType(x, } \textit{Hijacking}) \text{ hasAttackType(x, } \textit{HostageTaking}))$$

- A weapon of type chemical can be an explosive or poisoning weapon.

$$\exists y(\text{Weapon(y)} \wedge \text{Chemical(y)} \rightarrow \text{Explosive(y)} \vee \textit{Poisoning}(y))$$

- All incidents must have at least one weapon type recorded.

$$\forall x \exists y(\text{Incident (x)} \wedge \text{Weapon(y)} \rightarrow \text{hasWeaponType(x, y)})$$

- For some incidents, more specific sub-weapon types can be recorded.

$$\exists x \exists y(\text{Incident (x)} \wedge \text{Weapon(y)} \wedge \text{Chemical(y)} \wedge (\text{Explosive(y)} \vee \textit{Poisoning}(y) \rightarrow \text{hasWeaponType(x, y)} \wedge \text{hasWeaponSubType(x, y)}$$

3.5. Implementation Phase

The GTDOnto ontology is implemented in OWL/RDF using Protégé. The GTDOnto ontology contains 251 classes, 232 subsumption relations, 20 object properties, and 58 data properties, as shown in Figure 2. In addition, the ontology includes 29 individuals of different class types for evaluation purposes. The results and evaluation section explains the detailed description and analysis of this ontology's components.

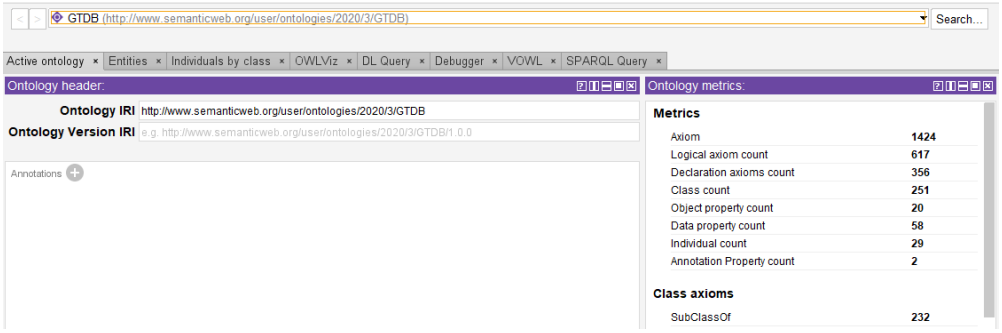

Figure 2. GTDOnto ontology metrics in Protégé.

All the concepts depicted in the previous phases are built as classes in protégé for the GTDOnto. Each class was assigned its object properties and data properties as conceptualized. Figure 3 illustrates the object properties built in GTDOnto Ontology. Each object property has a domain and a range of different related classes. The data properties are used to describe literal values for classes. Figure 4 illustrates the GTDOnto implementation of these properties.

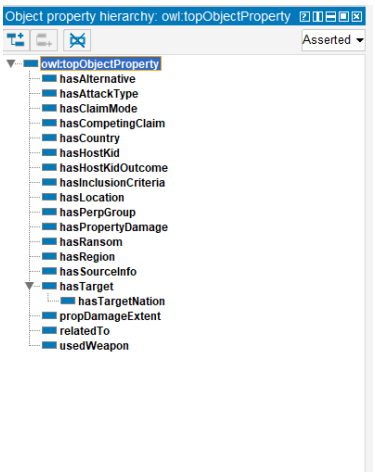

Figure 3. Object properties implementation.

Figure 4. Data properties implementation in GTDOnto in GTDOnto.

4. The GTDOnto Ontology

The developed GTDOnto ontology is available upon request from the authors, as this is an ongoing effort to advance this work. However, due to the scarcity of ontologies in the terrorism domain and the lack of documentation for some, GTDOnto does not reuse any existing ontology. Figure 5 represents the result of all the object properties linked to the incident class being its domain and other classes being its range. For example, the *hasAttackType* object property is represented by the edge connecting the *Incident* with the *Attack*. The *relatedTo* object property is represented by the loop edge on the *Incident* class.

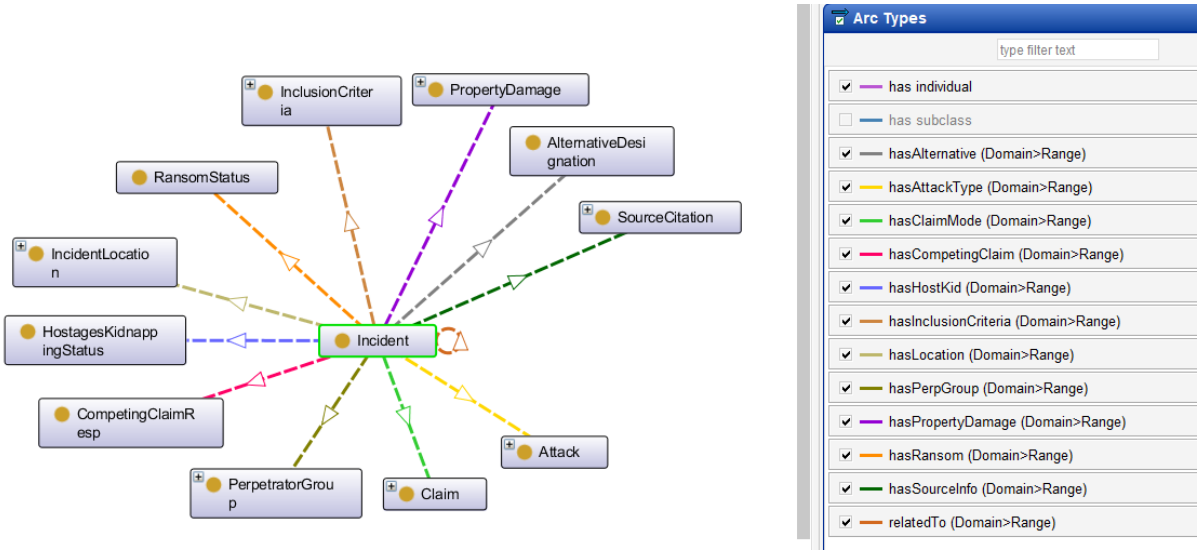

Figure 5. Incident class and its relations.

Figure 6 shows a snippet of the class hierarchy of the GTDOnto ontology implemented on Protégé. The incident class is highlighted in the hierarchy panel on the left side, and its related object properties are detailed on the right side of the figure. All the object properties connect the Incident class with other classes to record all the information about an incident.

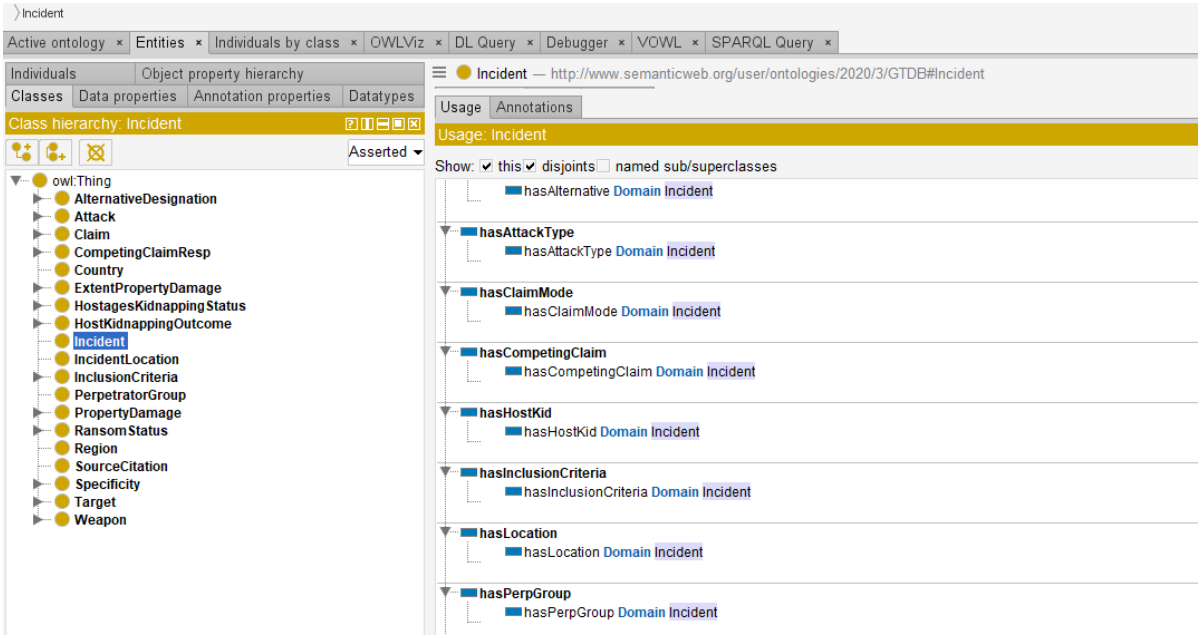

Figure 6. GTDOnto classes and the details of the Incident class.

The GTDOnto ontology models any knowledge about global incidents stored in the GTD. For example, Figure 7 details the Weapon class, including all its types and subtypes implemented as subsumption relation with the Weapon class—additionally, an object property relating Weapon with an incident that used a specific weapon type.

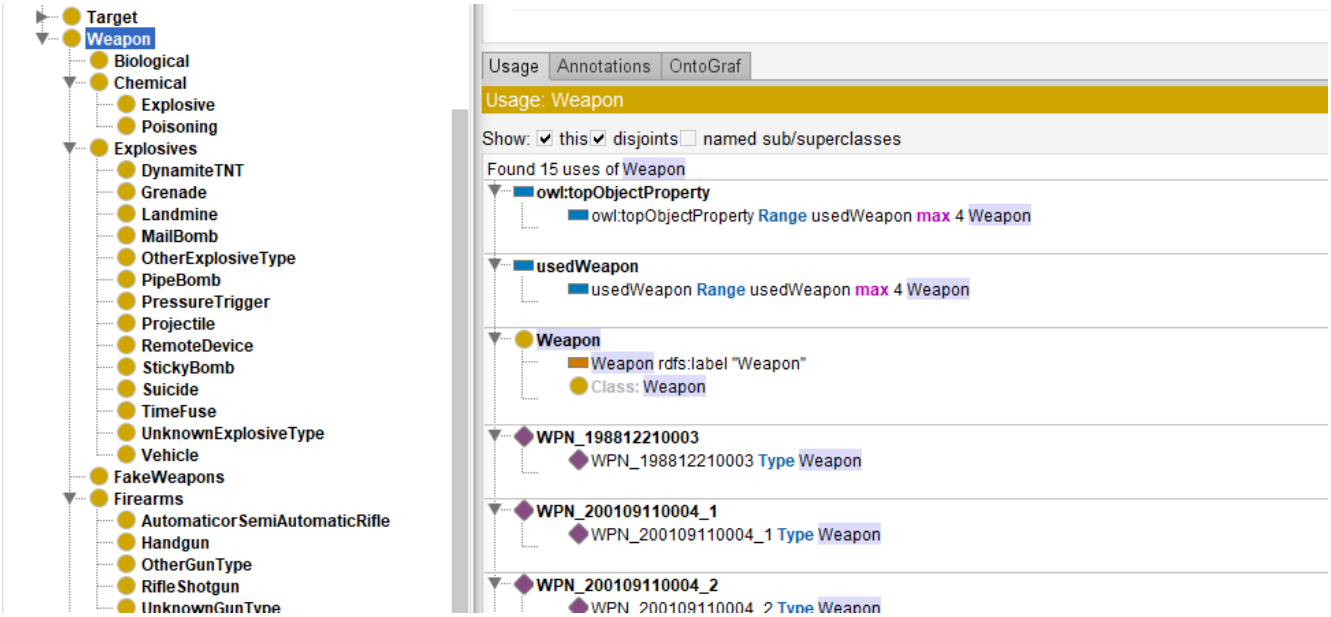

Figure 7. GTDOnto Weapon class details.

Furthermore, many details about an incident are recorded using data properties. For example, more information on the incident should be reported if an incident involved the hijacking or kidnapping of victims. Figure 8 shows an example of the many data properties for the class *VictimsHostageKidnapped*—these data properties record information about the incident victims who were taken as hostages or kidnapped.

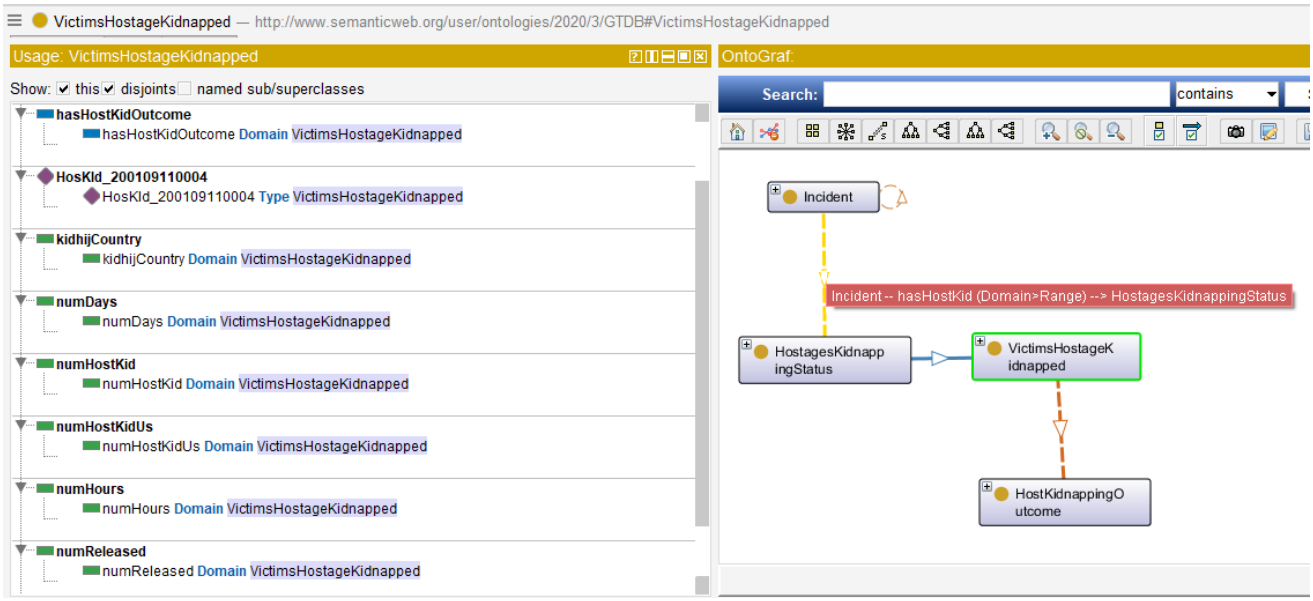

Figure 8. GTDOnto hostages or kidnapping details.

In addition to building classes and properties, cardinality restrictions are added to some properties to denote the number of maximum relations a class can have. For example, an incident can have multiple attack types in the same terrorist incident. Such restriction is represented in the GTDOnto object property "hasAttackType" which limits the maximum cardinality to three types of attacks for an incident. The sample OWL/RDF code details this restriction on the object property hasAttackType.

```
<!- http://www.semanticweb.org/user/ontologies/2020/3/GTDB#hasAttackType ->
   <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2020/3/GTDB#hasAttackType">
       <rdfs:subPropertyOf
rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
       <rdfs:domain
rdf:resource="http://www.semanticweb.org/user/ontologies/2020/3/GTDB#Incident"/>
       <rdfs:range>
<owl:Restriction> <owl:onProperty
rdf:resource="http:
//www.semanticweb.org/user/ontologies/2020/3/GTDB#hasAttackType"/>
<owl:maxQualifiedCardinality
rdf:datatype="http:
//www.w3.org/2001/XMLSchema#nonNegativeInteger">3</owl:maxQualifiedCardinality>
<owl:onClass
rdf:resource="http://www.semanticweb.org/user/ontologies/2020/3/GTDB#Attack"/>
           </owl:Restriction>
       </rdfs:range>
   </owl:ObjectProperty>
```

5. Evaluation and Discussion

Ontology evaluation aims to assess the developed ontology's quality and correctness, where the evaluation change according to the ontology development method [33]. The work of [34] summarized ontology evaluation approaches into four approaches: (1) gold standard approach that compares the developed ontology to an existing (gold standard) and finds similarities/differences; (2) data-driven approach evaluates against a given corpus (set of terms or documents); (3) the application-based approach evaluates the ontology in performing a specific task; and (4) criteria-based approaches such as human assessment, which asks humans (usually domain experts) to evaluate the ontology. Further evaluation is conducted on the schema design of the ontology based on the OntoQA evaluation tool [35] and presented in this section. In this work, the gold standard approach does not apply since no existing ontology exists to match. On the other hand, the data-driven approach can be used in future work to measure the usefulness of the GTDOnto with several applications developed to use it. Hence, the latter two approaches were used to evaluate the GTDOnto.

5.1. Running Examples

Based on the task-based approach, two instances of the incident class are created to evaluate the GTDOnto and assess if the ontology can be used to describe different incidents from the GTD and have all the terms and properties of that incident represented. Therefore, a couple of individuals were instantiated of type Incident class. Two incidents were selected from the GTD, and the information described in the database was taken as is and represented using our proposed ontology. Other types of individuals were created to represent all information about these incidents, as shown in Figures 9 and 10. The famous 1988 Lockerbie aircraft bombing incident is illustrated in Figure 9, and the September 11 incident is represented in Figure 10. The primary and first task for building GTDOnto is to represent the incidents in a graph structure that can be queried efficiently. The GTDOnto exploration is further investigated by answering several competency questions using SPARQL queries.

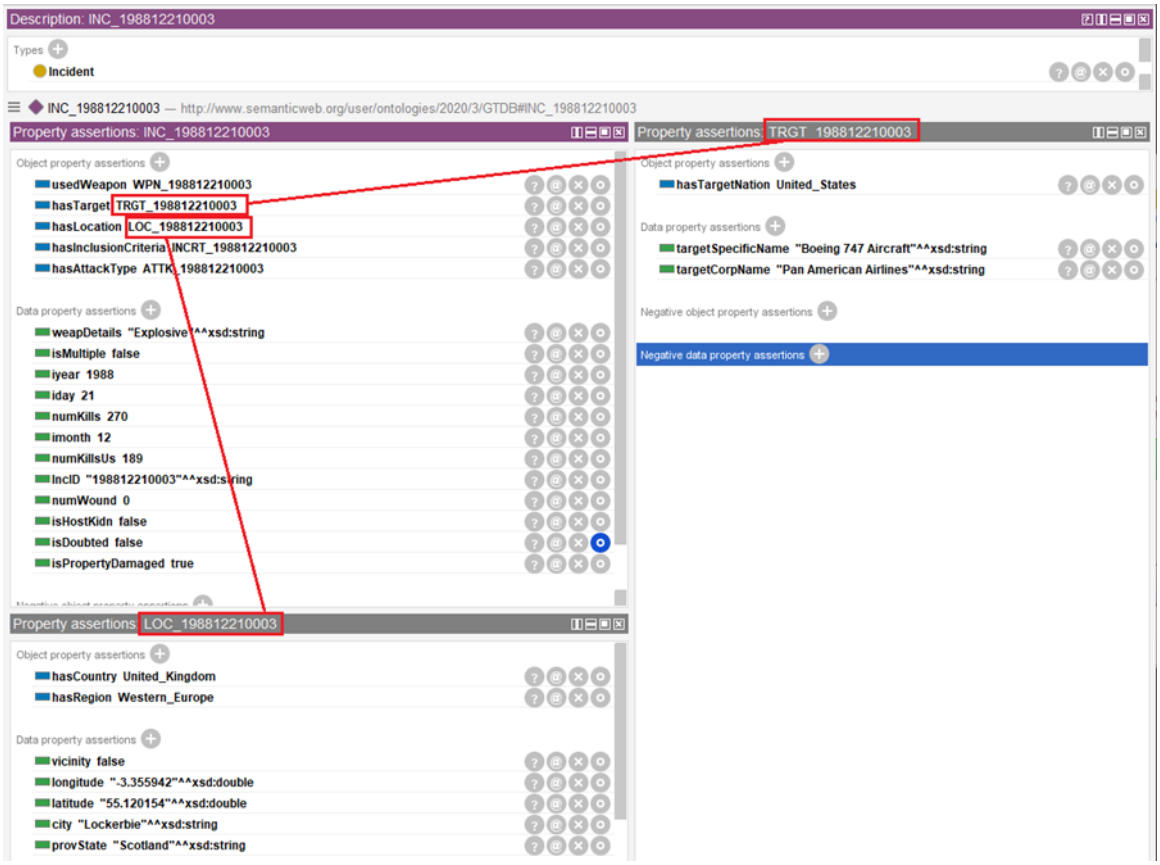

Figure 9. GTD incident (ID: 198812210003) representation in the GTDOnto (1988 Lockerbie incident).

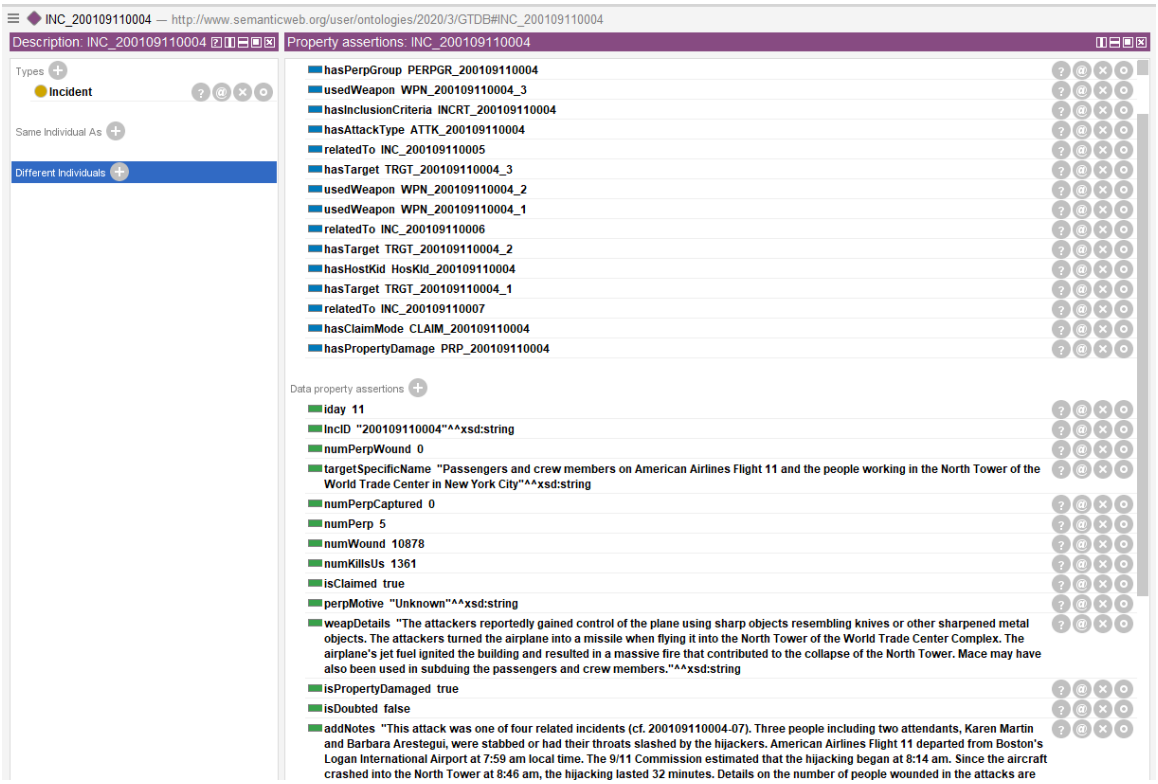

Figure 10. GTD Incident GTD (ID: 200109110004) representation in the GTDOnto (the September 11 incident).

The experiment for representing the above two incidents indicates that the GTDOnto can precisely represent all the properties describing incidents, whether they have detailed information or lack some information. Figure 9 details the data and object properties, with snippets of the other related individuals to the Lockerbie incident. This incident was used with 38 properties and relations—for example, the individuals of type location and target detail extra information about the incident. The GTDOnto representation for all incidents in this form will result in a knowledge graph of incidents and other class types. Figure 10 represents one of the September 11 incidents. It was used with 82 properties and relations. Full details about this incident were detailed in the GTD; hence, its representation in the GTDOnto is detailed. The object properties for this incident relate it to three different weapons used, three different targets, and three related incidents, as shown in the object properties.

5.2. Competency Questions Answering

The ontology covers the main concepts to describe an act of terrorism. Regarding the competency questions used to define the GTDOnto in the ORSD presented in Table 1, the ontology can answer all the competency questions asked in the functional requirements definition. Additionally, answering competency questions is part of the NeOn development methodology for the assessment of the GTDOnto ontology.

A sample of the competency questions from the groups specified in Table 1 is answered using SPARQL to evaluate the completeness criteria of the GTDOnto. Additionally, this evaluation assesses the correct representation of domains and ranges for the properties.

First, the Incident class, with all the associated data and object properties, can provide general answers about the dataset and much more detailed information about a specific incident. For example, the CQ1.1 (How many incidents occurred in the year 2005?) is resolved with the following SPARQL query:

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select (count(?inc) as ?IncNum) where {
   ?inc :iyear "2001"^^xsd:integer.
   ?inc :hasLocation ?loc.
   ?loc :hasCountry ?country.
   ?country rdfs:label "United States"^^xsd:string.
}
```

With the Attack class modeled and the data and object properties associated with it, it is possible to answer several competency questions about that concept. The CQ2.1 (What are the possible types of attacks associated with global incidents?) is resolved with the following SPARQL query:

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?attkTypes where {
   ?attkTypes rdfs:subClassOf :Attack.
}
```

The result of this query is all the types of attack associated with any act of terrorism as coded in the GTD codebook, e.g., armed assault, bombing explosion, assassination . . . , etc. Furthermore, the GTDOnto can answer CQ2.4 (What are the incidents that recorded "BombingExolosive" attacks?) to enlist all incidents of attack type Bombing explosion with the following SPARQL query:

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?inc where {
   ?inc :hasAttackType ?attk.
   ?attk rdf:type :BombingExplosion.
}
```

Furthermore, with the Weapon class modeled and the data and object properties associated with it, it is possible to answer more detailed questions. For example, CQ3.2 (What are the possible weapons subtypes of "explosives" weapons type?) enquires about subtypes of explosive weapons. It can be resolved with the following SPARQL query.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?explosiveTypes where {
   ?explosiveTypes rdfs:subClassOf :Explosives.
}
```

It is also possible to answer CQ3.3 (What types and subtypes of weapons were used in the incident with an ID "200109110004"?) about a specific incident with the following SPARQL query.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?wpn ?subWpn where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :usedWeapon ?w.
   ?w rdf:type ?wpn.
   ?wpn rdfs:subClassOf :Weapon.
   Optional {
       ?w rdf:type ?subWpn.
       ?subWpn rdfs:subClassOf ?wpn.  }
}
```

The result of this SPARQL query returns the following weapon types (subtypes): Incendiary (Gasoline Alcohol), Melee (Knife or Other Sharp Object), and Vehicle for the incident illustrated in Figure 10.

With the Targets class modeled and the data and object properties associated with it, it is possible to answer several competency questions, such as CQ4.3 (What are the nationalities of all targets/victims of a terrorist incident?) for a specific incident. Such questions can be resolved with the following SPARQL query.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
Select distinct ?nationality where {
   ?inc :hasTarget ?trgt.
   ?trgt :hasTargetNation ?country.
   ?country rdfs:label ?nationality.
}
```

It is possible to answer several conditional competency questions with information about perpetrators modeled in the Perpetrator class and the associated data and object properties. For example, to answer CQ5.1 (Does the terrorist incident claim responsibility by a group? If yes, what is the group name that carried out the incident?), the following SPARQL query is used.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?name ?claim where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :isClaimed ?claim.
   Optional {
       ?inc :hasPerpGroup ?group.
       ?group :perpGroupName ?name.  }
}
```

More descriptive information can be enquired about the casualties and consequences of the September 11 incident (Figure 10). For example, a simple question can be CQ6.1 (How many confirmed fatalities and injuries were reported in a terrorist incident?). A more detailed question is CQ6.4 (Were there any hostages in the incident? What is the outcome of reported hostage/kidnapping incidents? And what is the total number of hostages?). The CQ6.1 is resolved by the following SPARQL query to show the number of kills and wounded people for an incident.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?nKills ?nWounded where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :numKills ?nKills.
   ?inc :numWound ?nWounded.
}
```

While CQ6.4 considers the incident of hijacking a plane, the following SPARQL details the number of hostages, hours, and the outcome of this hijacking.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?numHostKilledk ?numHours ?hostageStatus where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :isHostKidn "true"^^xsd:boolean.
   ?inc :hasHostKid ?hostKid.
   ?hostKid :numHostKid ?numHostKilledk.
   ?hostKid :numHours ?numHours.
   ?hostKid :hasHostKidOutcome ?outcome.
   ?outcome rdf:type ?hostageStatus.
   ?hostageStatus rdfs:subClassOf :HostKidnappingOutcome.
}
```

This evaluation provided a sample of SPARQL used to answer all the competency questions in the functional requirements of the ORSD for the GTDOnto (Table 1).

5.3. GTDOnto Ontology Quality

The current ontology version utilizes all the concepts, relations, and properties described in the codebook for the GTD project maintained by START at the University of Maryland. Nevertheless, a logical evaluation was conducted by domain experts to verify the accuracy of the naming of concepts and to validate the hierarchy of the terms presented in GTDOnto ontology. Human assessment of an ontology evaluates its quality. Hence, experiments were conducted with experts from the University of Jordan Center of Strategic Studies (CSS). The center primarily studies and researches regional conflicts, international relations, and security.

The criteria described in [36] are assessed as part of the evaluation process. These criteria are listed below with an explanation of their application in GTDOnto.

- Accuracy: The ontology development was assisted by the guidelines from the GTD codebook. Furthermore, the ontology evaluation was assessed by domain experts from JCSS. Classes and relations were evaluated regarding the accuracy of terms developed in the GTDOnto, compared to the terms and concepts of the GTD codebook. The assessment was performed via a questionnaire sent to the experts via email, and further meetings were conducted to demo the GTDonto.
- Adaptability: Each concept in the GTDOnto is represented using URIs and can be reused by other linked datasets when published. Thus, the GTDOnto can be reused and extended easily, making it adaptable.
- Clarity: All the classes, subclasses, and properties names defined in the GTDOnto are non-ambiguous names and ease human readability, which facilitates the creation of individuals of incidents and their related concepts without confusion. Experiments with domain experts from the CSS assessed the clarity by comparing the incidents described in the GTD excel sheet to the incident representation in the GTDOnto (Figures 9 and 10).
- Completeness: The GTDOnto ontology can answer all the competency questions specified in the functional requirements presented in the ORSD document (Table 1). Details for answering these competency questions are discussed in the next section.
- Efficiency: Creating instances is simple due to the clarity of naming and relations between concepts. Furthermore, the process of querying the GTDOnto is seamless with the sample of individuals created. However, further investigation should be carried out with a more significant number of individuals.
- Consistency: No inconsistencies were found in GTDOnto after performing reasoning using HermiT 1.4.3.456 reasoner.

5.4. Metric-Based Ontology Evaluation

To provide further evaluation for GTDOnto, the schema-based metrics that address the design of the ontology provided by the OntoQA evaluation tool [35] are addressed below:

- **The number of classes**: The total number of classes in GTDOnto is 251, as indicated in Figure 3.
- **The number of properties**: The number of properties in GTDOnto is 78, the combination of data and object properties describing all the classes' attributes and their relations.
- **The number of root classes**: Despite having the Incident class as the main class for describing terrorist events, the GTDOnto has 19 root classes, indicating that the ontology is comprehensive and describes several concepts related to a global incident in its design.
- **Relationship Richness (RR)**: The relationship richness is represented as the percentage of the sub-class relationships between classes compared to all the possible connections between the ontology classes. It is computed as the ratio of the number of (non-inheritance) relationships (P), divided by the total number of relationships defined in the schema, where (P) is the number of (non-inheritance) relationships and (H) is the number of inheritance relationships using the equation

$$RR = |P|/(|H| + |P|)$$

  In GTDOnto, the RR is around 0.08 due to many subclasses in the GTDOnto schema.

- **Inheritance Richness (IR)**: Inheritance richness is a good indication of how well knowledge is grouped into different concepts in the ontology. The IR is defined as the average number of subclasses per class (C):

$$IR = |H|/|C|$$

In GTDOnto, the IR is around 0.9 since the ontology describes several concepts related to terrorist incidents, such as Attack types, Target types, Weapon types, and others.

- **Attribute Richness (AR)**: The number of attributes defined for each class indicates the amount of information conveyed describing incidents. The AR is calculated as the average number of attributes per class. It is computed as the number of attributes for all classes (att) divided by the number of classes (C):

$$\mathrm{AR} = |\mathrm{att}|/|\mathrm{C}|$$

The AR in GTDOnto is around 0.32. In the previous section, visualizing instances of class *Incident* indicates how much knowledge is conveyed to the Incident class but not to other classes, such as *Attack*, *Weapon*, and *Target*.

Evaluating GTDOnto proved that it satisfied its main goal: to represent all the information about any incident in the GTD in a machine-readable format. The evaluation focused on assessing the applicability of GTDOnto based on representing GTD incidents in a task-based evaluation followed by competency questions answering. This evaluation proved that all the details about any terrorism incident are covered with GTDOnto representation. Furthermore, evaluation based on human assessment and competency questions answering assured that this work covers the information used to describe any terrorism incident in detail. Finally, schema-based evaluation of the GTDOnto focused on the design showed that the GTDOnto covers vast concepts related to the main class, which is the *Incident* class in the GTDOnto.

The development of GTDOnto is an ongoing effort. One goal is to keep the GTDOnto updated to cover all the classes and subclasses in the GTD database as the database update regularly. Furthermore, this GTDOnto is available for download for future efforts of researchers and developers to enhance. The uploaded ontology contains two instances from the GTD provided as examples that can be queried using SPARQL. In its current version, the GTDOnto can be the base for several applications in the future. We consider using the GTDOnto as the base for building a knowledge graph representing all the terrorism incidents of GTD, which will be helpful for further analysis tasks. We envision the GTDOnto expanding by incorporating rules to classify terrorism incidents of similar types.

Furthermore, GTDOnto terms and relations can be incorporated into tools for annotating content published on the web that might indicate terrorist-like intentions. Other than that, in the near future, we aim to reach out and work with the GTD project to integrate the GTDOnto with their efforts and publish the dataset in a machine-readable format. Furthermore, we hope publishing this dataset as part of the linked open data cloud will leverage the work and connect it with other existing datasets about the media or social media incidents.

6. Conclusions

The Global Terrorism Database (GTD) is made available by organizations such as the National Consortium for the Study of Terrorism and Responses to Terrorism (START). Experts in the domain gather this dataset, but technical modeling for its metadata is lacking. Hence, based on the guidelines of Scenario 1: from specification to implementation, from the NeOn ontology development methodology, we designed the GTD Ontology (GTDOnto). The aim was to model the incidents, targets, attackers, weapons, and other associated information and organize the knowledge on terrorism. Expanding on the work of START, this project aims to provide controlled vocabularies in a machine-readable, interoperable format, thereby establishing a conceptual model that can be utilized and expanded to characterize potential instances.

Furthermore, evaluation based on running examples, human assessment, and competency questions answering was undertaken to verify the utility of the developed ontology. Hence, future work will expand the GTDOnto to infer types of incidents based on specific criteria and examine the use of GTDOnto as an underlying schema to build a knowledge graph for

terrorism. The work is an ongoing effort that we hope can be further used to serve researchers in this field for enhanced research, such as prediction and other downstream tasks.

Author Contributions: Conceptualization, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; methodology, R.Q.A.-F. and M.A.-T.; software, R.Q.A.-F. and M.A.-T.; validation, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; formal analysis, R.Q.A.-F. and M.A.-T.; data curation, R.Q.A.-F. and M.A.-T.; writing—original draft preparation, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; writing—review and editing, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; visualization, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: GTDOnto Ontology. Triplydb. https://triplydb.com/ReemAlfayez/GTDOnto/, accessed on 12 January 2020.

Acknowledgments: We would like to thank the Centre of Strategic Studies at the University of Jordan for sharing their expertise to review and evaluate the work presented in this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Inkpen, R.; Gauci, R.; Gibson, A. The Values of Open Data. *Area* **2021**, *53*, 240–246. [CrossRef]
2. Mendes, P.S.F.; Siradze, S.; Pirro, L.; Thybaut, J.W. Open Data in Catalysis: From Today's Big Picture to the Future of Small Data. *ChemCatChem* **2021**, *13*, 836–850. [CrossRef]
3. Ruijer, E.; Grimmelikhuijsen, S.; van den Berg, J.; Meijer, A. Open Data Work: Understanding Open Data Usage from a Practice Lens. *Int. Rev. Adm. Sci.* **2018**, *86*, 3–19. [CrossRef]
4. Stieglitz, S.; Wilms, K.; Mirbabaie, M.; Hofeditz, L.; Brenger, B.; López, A.; Rehwald, S. When Are Researchers Willing to Share Their Data?—Impacts of Values and Uncertainty on Open Data in Academia. *PLoS ONE* **2020**, *15*, e0234172. [CrossRef] [PubMed]
5. Pietriga, E.; Gözükan, H.; Appert, C.; Destandau, M.; Čebirić, Š.; Goasdoué, F.; Manolescu, I. Browsing Linked Data Catalogs with LODAtlas. In Proceedings of the International Semantic Web Conference, Monterey, CA, USA, 8–12 October 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 137–153. [CrossRef]
6. Alvite-Díez, M.L. Linked Open Data Portals: Functionalities and User Experience in Semantic Catalogues. *Online Inf. Rev.* **2021**, *45*, 946–963. [CrossRef]
7. Bizer, C.; Heath, T.; Berners-Lee, T. Linked Data the Story so Far. In *Semantic Services, Interoperability and Web Applications: Emerging Concepts*; IGI global: Hershey, PA, USA, 2011; pp. 205–227.
8. Park, S.; Gil-Garcia, J.R. Open Data Innovation: Visualizations and Process Redesign as a Way to Bridge the Transparency-Accountability Gap. *Gov. Inf. Q.* **2022**, *39*, 101456. [CrossRef]
9. Donaldson, D.R.; Koepke, J.W. A Focus Groups Study on Data Sharing and Research Data Management. *Sci. Data* **2022**, *9*, 1–7. [CrossRef] [PubMed]
10. Chapman, A.; Simperl, E.; Koesten, L.; Konstantinidis, G.; Ibáñez, L.D.; Kacprzak, E.; Groth, P. Dataset Search: A Survey. *VLDB J.* **2019**, *29*, 251–272. [CrossRef]
11. Brickley, D.; Burgess, M.; Noy, N. Google Dataset Search: Building a Search Engine for Datasets in an Open Web Ecosystem. In Proceedings of the World Wide Web Conference, WWW 2019, San Francisco, CA, USA, 13–17 May 2019; Association for Computing Machinery, Inc.: New York, NY, USA, 2019; pp. 1365–1375. [CrossRef]
12. Charalabidis, Y.; Zuiderwijk, A.; Alexopoulos, C.; Janssen, M.; Lampoltshammer, T.; Ferro, E. Open Data Interoperability. In *The World of Open Data. Public Administration and Information Technology*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 75–93. [CrossRef]
13. Mannes, A.; Golbeck, J. Building a Terrorism Ontology. In Proceedings of the ISWC Workshop on Ontology Patterns for the Semantic Web, Galway, Ireland, 6–10 November 2005.
14. Mannes, A.; Golbeck, J. Ontology Building: A Terrorism Specialist's Perspective. In Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2007; pp. 1–5.
15. Inyaem, U.; Haruechaiyasak, C.; Meesad, P.; Tran, D. Ontology-Based Terrorism Event Extraction. In Proceedings of the 2009 1st International Conference on Information Science and Engineering, Nanjing, China, 26–28 December 2009; pp. 912–915. [CrossRef]
16. START (National Consortium for the Study of Terrorism and Responses to Terrorism). Global Terrorism Database 1970–2020. Available online: https://start.umd.edu/gtd/ (accessed on 15 July 2022).
17. LaFree, G.; Dugan, L. Introducing the Global Terrorism Database. *Terror. Political Violence* **2007**, *19*, 181–204. [CrossRef]
18. Shadbolt, N.; Hall, W.; Berners-Lee, T. The Semantic Web Revisited. *IEEE Intell. Syst.* **2006**, *21*, 96–101. [CrossRef]
19. Gruber, T.R. A Translation Approach to Portable Ontology Specifications. *Knowl. Acquis.* **1993**, *5*, 199–220. [CrossRef]
20. Aminu, E.F.; Oyefolahan, I.O.; Abdullahi, M.B.; Salaudeen, M.T. A Review on Ontology Development Methodologies for Developing Ontological Knowledge Representation Systems for Various Domains. *Int. J. Inf. Eng. Electron. Bus.* **2020**, *2*, 28–39. Available online: https://www.mecs-press.org/ijieeb/ijieeb-v12-n2/IJIEEB-V12-N2-5.pdf (accessed on 10 January 2023). [CrossRef]

21. Uschold, M.; King, M. *Towards a Methodology for Building Ontologies*; Presented at "Workshop on Basic Ontological Issues in Knowledge Sharing"; Artificial Intelligence Applications Institute, University of Edinburgh: Edinburgh, UK, 1995.
22. Kotis, K.; Vouros, G.A. Human-Centered Ontology Engineering: The HCOME Methodology. *Knowl. Inf. Syst.* **2006**, *10*, 109–131. [CrossRef]
23. Noy, N.F.; McGuinness, D.L. Ontology Development 101: A Guide to Creating Your First Ontology. 2001. Available online: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf (accessed on 10 December 2022).
24. Denaux, R.; Dolbear, C.; Hart, G.; Dimitrova, V.; Cohn, A.G. Supporting Domain Experts to Construct Conceptual Ontologies: A Holistic Approach. *J. Web Semant.* **2011**, *9*, 113–127. [CrossRef]
25. Suárez-Figueroa, M.C.; Gómez-Pérez, A.; Fernández-López, M. The Neon Methodology for Ontology Engineering. In *Ontology Engineering in a Networked World*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 9–34. [CrossRef]
26. Suárez-Figueroa, M.C. NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse. Ph.D. Thesis, Universidad Politécnica de Madrid facultad de informática, Madrid, Spain, 2010.
27. Zemmouchi-Ghomari, L.; Ghomari, A.R. Process of Building Reference Ontology for Higher Education. In Proceedings of the World Congress on Engineering: WCE, London, UK, 3–5 July 2013; pp. 1595–1600.
28. Fonou-Dombeu, J.V.; Achary, T.; Genders, E.; Mahabeer, S.; Pillay, S.M. COVIDonto: An Ontology Model for Acquisition and Sharing of COVID-19 Data. In Proceedings of the International Conference on Model and Data Engineering, Tallinn, Estonia, 21–23 June 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 227–240. [CrossRef]
29. Ibanescu, L.; Dibie, J.; Dervaux, S.; Guichard, E.; Raad, J. PO2—A Process and Observation Ontology in Food Science. Application to Dairy Gels. In *Metadata and Semantics Research. MTSR 2016. Communications in Computer and Information Science*; Garoufallou, E., Subirats Coll, I., Stellato, A., Greenberg, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; Volume 672, pp. 155–165. [CrossRef]
30. Suárez-Figueroa, M.C.; Gómez-Pérez, A.; Fernández-López, M. The NeOn Methodology Framework: A Scenario-Based Methodology for Ontology Development. *Appl. Ontol.* **2015**, *10*, 107–145. [CrossRef]
31. START. Global Terrorism Database Codebook: Inclusion Criteria and Variables. 2019. Available online: https://www.start.umd.edu/gtd/downloads/Codebook.pdf (accessed on 23 July 2022).
32. Suárez-Figueroa, M.C.; Gómez-Pérez, A.; Villazón-Terrazas, B. How to Write and Use the Ontology Requirements Specification Document. In *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5871, pp. 966–982.
33. Sabou, M.; Fernandez, M. Ontology (Network) Evaluation. In *Ontology Engineering in a Networked World*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 193–212.
34. Raad, J.; Cruz, C. A Survey on Ontology Evaluation Methods. In Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Lisbon, Portugal, 12–14 November 2015.
35. Tartir, S.; Arpinar, I.B.; Sheth, A.P. Ontological Evaluation and Validation. In *Theory and Applications of Ontology: Computer Applications*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 115–130. [CrossRef]
36. Vrandečić, D. Ontology Evaluation. In *Handbook on Ontologies*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 293–313. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.