FUZRUF-onto: A Methodology to Develop Fuzzy Rough Ontologies

Rawan Sanyour[1,*], Manal Abdullah[2]

[1,2] *Information Systems Department, Faculty of Computing and Information Technology, King AbdulAziz University, Jeddah, Saudi Arabia*

(Received: November 13, 2024; Revised: December 19, 2024; Accepted: January 17, 2025; Available online: March 15, 2025)

Abstract

Nowadays, semantic web technologies play a crucial role in the knowledge representation paradigm. With the rise of imprecise and vague knowledge, there is an upsurge demand in applying a concrete well-established procedure to represent such knowledge. Ontologies, particularly fuzzy ontologies, are increasingly applied in application scenarios in which handling of vague knowledge is significant. However, such fuzzy ontologies utilize fuzzy set theory to provide quantitative methods to manage vagueness. In various cases of real-life scenarios, people need to express their everyday requirements using linguistic adverbs such as very, exactly, mostly, possibly, etc. The aim is to show how fuzzy properties can be complemented by Rough Set methods to capture another type of imprecision caused by approximation spaces. Rough sets theory offers a qualitative approach to model such vagueness via describing fuzzy properties at multiple levels of granularity using approximation sets. Using rough-set theory, each fuzzy concept is represented by two approximations. The lower approximation PL(C) consists of a set of fuzzy properties that are definitely observable in the concept. The upper approximation PU(C) on the other hand contains fuzzy properties that are possibly associated with the concept but may not be observed. This paper introduces a methodology named FUZRUF-onto methodology, which is a formal guidance on how to build fuzzy rough ontologies from scratch using extensive research in the area of fuzzy rough combination. Fuzzy set and rough set theories are applied to capture the inherently fuzzy relationships among concepts expressed by natural languages. The methodology provides a very good guideline for formally constructing fuzzy rough ontologies in terms of completeness, correctness, consistency, understandability, and conciseness. To explain how the FUZRUF-onto works, and demonstrate its usefulness, a practical step by step example is provided.

Keywords: Fuzzy Sets, Fuzzy Theory, Rough Sets, Fuzzy Ontologies, Ontology Engineering, Knowledge Representation

1. Introduction

Ontologies provide an explicit representation of generalized conceptualizations and have been recognized as one of the most effective techniques for modeling and representing knowledge [1], [2], [3]. They have been widely applied in diverse fields such as semantic web, soft computing, artificial intelligence, software engineering, and natural language processing [4]. According to [5], ontologies can be defined as explicit formal specifications that represent real-world concepts and relationships within a specific domain, enabling the establishment of interrelationships with other models in an automated manner.

Despite their success, classical ontologies are often inadequate for semantically representing vague and imprecise knowledge commonly encountered in real-world domains [6]. Vagueness, expressed through everyday terms like close, hot, tall, very, or roughly, is prevalent in human language but lacks sharp boundaries between entities. This imprecision creates challenges in certain domains where concepts exhibit fuzzy boundaries that cannot be precisely defined. Representing such vagueness in ontologies is crucial not only because it exists across many domains but also because addressing vagueness can significantly enhance a system's effectiveness in various application scenarios [7], [8].

Fuzzy set theory, rough set theory, and fuzzy logic [9], [10], [11], [12] have proven to be effective formalisms for managing vague and imprecise knowledge in real-world contexts. Fuzzy rough ontologies represent a new paradigm for knowledge representation, particularly useful in applications where vague or imprecise information plays a significant role. This paper introduces a novel methodology called FUZRUF-onto, designed to construct reusable and sharable fuzzy rough ontologies from scratch. The proposed methodology enhances the development process by

***Corresponding author: Rawan Sanyour (Rsanyour0001@stu.kau.edu.sa)**
DOI: https://doi.org/10.47738/jads.v6i2.625

improving the accuracy, shareability, and reusability of the resulting ontology. It provides a structured guideline to: (i) identify vague knowledge within a specific domain, and (ii) explicitly and accurately model this knowledge using fuzzy rough ontology elements.

While the development of fuzzy ontologies has been explored, methodological aspects remain underdeveloped. Previous works, such as FODM [13] and IKARUS-Onto [14], offer step-by-step guidelines for constructing fuzzy ontologies. These methodologies share common steps, such as establishing the need for fuzziness, identifying vagueness within the domain, defining fuzzy elements, and quantifying them using fuzzy degrees. Reusing existing ontologies is another key step, allowing developers to save time and effort by adapting available ontologies to meet new requirements. The final steps—formalizing and validating the ontology—ensure that the constructed ontology accurately captures and formalizes the intended knowledge.

The proposed FUZRUF-onto methodology integrates fuzzy set theory, which addresses vagueness, with rough set theory to manage uncertainty in data. Using rough set theory, the vagueness of a fuzzy concept is approximated through two properties: fuzzy lower and fuzzy upper bounds. This integration is highly effective for various applications. For example, in natural language, adjectives such as good, tall, or young describe objects' properties and can be represented using fuzzy sets. Rough set theory further models the vagueness of such adjectives qualitatively. Linguistic hedges, also called modifiers, are employed to describe vague properties of fuzzy concepts. These modifiers can intensify (e.g., very, extremely, definitely) or weaken (e.g., more or less, quite, rather) the emphasis on fuzzy properties.

FUZRUF-onto is the first methodology to combine fuzzy set and rough set theories, enabling precise identification of uncertainty in context-dependent characteristics of objects. This allows user preferences to be more accurately expressed by adjusting the truth levels of inferred contexts. The methodology offers a systematic approach to developing fuzzy rough ontologies, significantly improving the representation and management of vagueness.

2. Preliminaries

2.1. Fuzzy Logic and fuzzy sets

Fuzzy logic and fuzzy set theory were firstly introduced by Zadeh [15] to address imprecise and vague knowledge. In contrast to the classical set theory in which elements either belong to a specific set or not, in fuzzy sets, an element can be a set member to some degree. For example, if X is a set of elements, a fuzzy subset A of X can be defined by a membership function μA(X) that assign any $x \in X$ to a value in the interval of real numbers between 0 and 1. An element has a 0 value means that no membership while an element with 1 value represents a full membership. This change in the classical true/false convention has led to a new type of propositions called fuzzy propositions. Each of these propositions can have a degree of truthiness belong to [0,1]. Such degree reflects the compatibility of a fuzzy proposition with a given state of facts. For instance, the truth of a proposition saying that a given person is tall is clearly a matter of degree. All crisp set operations such as intersection, union, complement and implication set are extended to fuzzy sets using t-norm function, a t-conorm function, a negation function and an implication function respectively. For a formal definition of these functions, we refer the reader to [16], [17].

2.2. Rough Sets and Fuzzy Rough Sets

Fuzzy set theory provides a quantitative way to represent vagueness in knowledge using degrees of membership to fuzzy concepts. On the contrary, rough set theory offers a qualitative approach to manage this vagueness. Instead of using a degree of membership, rough sets are used to approximate vague concepts. It is an effective way when it is not possible to quantify the membership function of these vague concepts. Rough sets were firstly introduced by Pawlak in 1982 [18]. Using rough theory, the vagueness of a fuzzy concept can be approximated with two properties: fuzzy lower degree and fuzzy upper degree. This integration is useful in several domains of application. Often, in natural language, objects' properties are expressed using adjectives such as good, tall, young, etc. which can be represented using fuzzy set theoretical approach. Rough set theory offers a qualitative approach to model the vagueness of such adjectives. Linguistic hedges (also called modifiers) can be considered as special linguistic expressions to describe the vague properties of fuzzy concepts. Generally, linguistic hedges (modifiers) are special linguistic expressions by which an emphasis can be imposed on the corresponding fuzzy properties. They can be classified into two categories, i.e., the intensive hedges such as very, extremely, definitely, etc. that strengthen the emphasis imposed on the fuzzy term they are applied to, and the weakness hedges such as more or less, quiet, rather, etc. that weaken the emphasize. When there are only few elements belong to the concept, and there is an indiscernibility equivalence relation, i.e., reflexive, symmetric, and transitive relation between elements, a vague concept can be approximated by means of two concepts: Lower approximation and upper approximation. The lower approximation is defined by the sets of elements that definitely belong to the vague set. These elements may not be complete and not include some of the elements and features and contain all the indistinguishable elements of the vague set. On the other hand, the upper approximation describes the set of elements that possibly belong to the vague set, i.e., elements that might not actually belong to the vague concept. This set may consist of some elements that are indistinguishable from the vague set. A rough set then can be defined as a pair of these two approximations: lower and upper approximations.

Fuzzy logic and rough logic together can be considered as complementary formalisms to address impreciseness and vagueness in knowledge representation. An extension of the rough sets named fuzzy rough sets [10], [12], [19] can be considered through defining a fuzzy similarity relation instead of the indiscernibility equivalence relation between elements. While in rough sets an element can only belong to one equivalent class of similar elements, in fuzzy rough sets, an element can belong to several fuzzy similarity classes with different degrees of truth. There are some other fuzzy sets approximations such as tight approximation in which all the existing fuzzy similarity classes are included, and loose approximation that considers the best one among the similarity classes.

3. FUZRUF-Onto Methodology

The FUZRUF-Onto methodology is a structured approach to represent vague knowledge using fuzzy set and rough set theories. This methodology is designed to assist ontology engineers and domain experts in modeling domain vagueness effectively. It comprises six defined phases, illustrated in figure 1, each outlining specific actions and roles for participants.

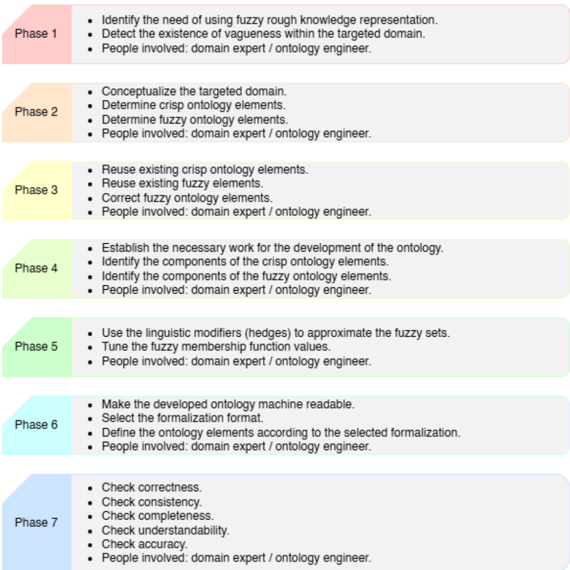

Figure 1. The FUZRUF-onto Lifecycle

3.1. Phase 1: Identifying Ontology Purpose and Motivation

The first phase involves identifying the need for a fuzzy rough ontology by evaluating the extent of vagueness and roughness in the target domain. This step defines the purpose and scope of the ontology and determines the necessary effort to develop it. The domain and scope of the knowledge to be modeled are established, and the type of ontology—whether domain-specific, application-specific, or generic—is determined. Roles and responsibilities of the domain experts and ontology engineers are assigned.

Domain experts contribute knowledge to identify vagueness, while ontology engineers ensure that the vagueness is correctly captured and appropriately represented. Vagueness in the domain is analyzed, focusing on elements such as vague concepts (e.g., young, adult) [20], vague relations (e.g., near, far) [21], and vague attributes (e.g., tall, short) [22]. This phase confirms the necessity of creating a fuzzy rough ontology.

3.2. Phase 2: Determining Ontology Elements

In this phase, ontology elements are identified and categorized into crisp and fuzzy components. Crisp elements, such as classes, properties, and relationships, are defined using existing methodologies and organized hierarchically [20]. Fuzzy elements represent vague domain knowledge and are defined using fuzzy sets and membership functions. For example, fuzzy sets are represented mathematically as:

$$\mu_{A(x)}: X \rightarrow [0,1] \tag{1}$$

$\mu_{A(x)} = 1$ indicates full membership, $\mu_{A(x)} = 0$ indicates no membership, and values between 0 and 1 represent partial membership [21], [22]. Domain experts provide precise specifications, while ontology engineers formalize these fuzzy elements.

3.3. Phase 3: Reusing Existing Ontologies

Reusing existing ontology elements reduces workload and ensures compatibility with other applications. Crisp ontologies from repositories such as Swoogle or W3C Wiki are evaluated for relevance [20], [23]. Fuzzy and fuzzy rough ontologies are also assessed for potential reuse, although they are less common and more specialized [24]. Domain experts refine and adapt inherited elements to align with new requirements. For instance, a fuzzy datatype such as *YoungAge*, defined as 20–40 years in an existing ontology, might be adjusted to fit the context of the new ontology [25]. Figure 2 provides an example of how fuzzy concepts and their properties are structured in the ontology.

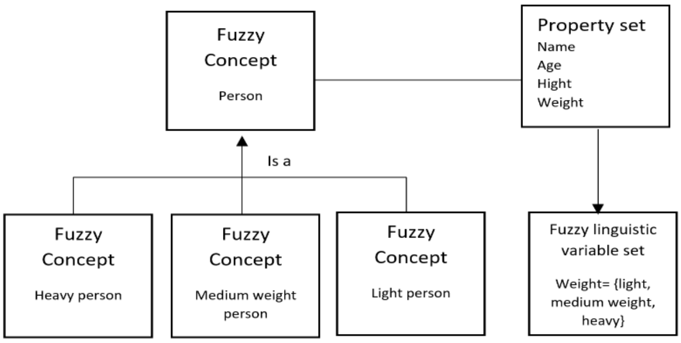

Figure 2. Example of a Fuzzy Concept with its Fuzzy Property

3.4. Phase 4: Defining Ontology Elements

This phase involves defining ontology elements to distinguish between crisp and fuzzy components. Fuzzy concepts are modeled based on their degrees of membership, which can be represented by equations such as:

$$\mu_{C(x)} = \max\left(0, 1 - \frac{|x-c|}{\sigma}\right) \tag{2}$$

c is the center of the fuzzy set, and c controls its spread [23], [24]. Fuzzy relations are represented using linguistic variables, as shown in figure 3, to define how instances of concepts relate to each other based on fuzzy degrees [25]. Additionally, figure 4 depicts an example of membership functions for the linguistic variable Age, dividing it into fuzzy categories such as Young, Middle-aged, and Old. Membership degrees (μ) show the extent to which an individual belongs to each category, providing a precise representation of vague information.

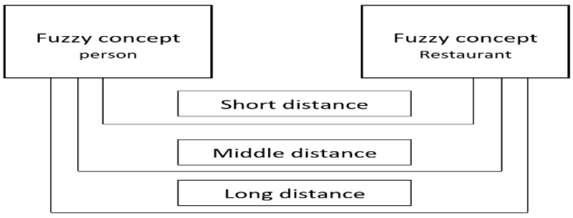

Figure 3. Fuzzy Relation with its Datatypes

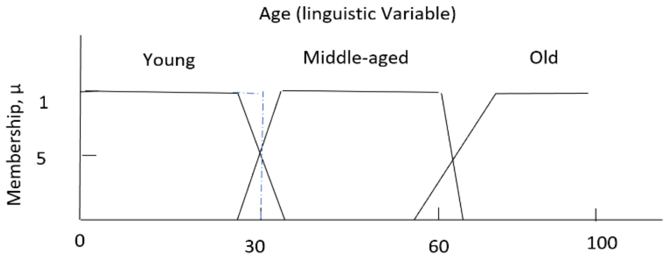

Figure 4. Fuzzy Property with its Datatypes

3.5. Phase 5: Approximating Fuzzy Concepts Using Linguistic Modifiers

Linguistic modifiers, such as very, extremely, and roughly, are applied to refine fuzzy concepts and sets. These modifiers adjust the membership values of fuzzy elements, allowing for a more nuanced representation of user preferences. Modifications are achieved through mathematical transformations, such as:

$$\mu_{m(A)}(x) = [\mu_A(x)]^p \tag{3}$$

$p > 1$ intensifies membership (e.g., very), and $0 < p < 1$ weakens it (e.g., somewhat) [26], [27], [28].

The role of fuzzy rough sets in approximating linguistic modifiers is detailed in table 1, which illustrates the relationship between modifiers, fuzzy rough sets, and membership functions [29], [30].

Table 1. Examples of Linguistic Modifiers with their Corresponding Fuzzy Rough approximations and Membership

Modifier	Fuzzy Rough Set	Membership Function
Very, exactly, definitely, certainly	Lower approximation	$(R_*A)(x) = inf_{y\in X} \max\{1 - R(x,y), A(y)\}$
Almost, more/less, rather, possible, potential, somewhat	Upper approximation	$(R^*A)(x) = inf_{y\in X} \min\{R(x,y), A(y)\}$
Extremely	Tight lower approximation	$(R_{\downarrow\downarrow}A)(x) = inf_{z\in X}\, inf_{y\in X} \max\{1 - R(x,z), \max\{1 - R(z,y), A(y)\}$
Very very	Loose lower approximation	$(R_{\uparrow\downarrow}A)(x) = sup_{z\in X}\, inf_{y\in X} \max\{1 - R(x,z), \max\{1 - R(z,y), A(y)\}$
Indeed	Tight upper approximation	$(R^{\uparrow\downarrow}A)(x) = inf_{z\in X}\, sup_{y\in X} \min\{R(x,z), \min\{R(z,y), A(y)\}$
Nearly, roughly	Loose upper approximation	$(R^{\uparrow\uparrow}A)(x) = sup_{z\in X}\, sup_{y\in X} \min\{R(x,z), \min\{R(z,y), A(y)\}$

3.6. Phase 6: Formalizing the Constructed Ontology

The fuzzy rough ontology is transformed into a machine-readable format in this phase using specialized languages like fuzzy OWL 2 [31]. These languages support the representation of vague and imprecise knowledge through features such as fuzzy datatypes and linguistic variables [32], [33]. The ontology engineers evaluate the reasoning capabilities of these languages to ensure they align with the domain's requirements and adequately support inference processes [34].

3.7. Phase 7: Validating the Constructed Ontology

Validation ensures that the fuzzy rough ontology achieves its intended goals. The constructed ontology is checked for correctness, ensuring it reflects the domain accurately [35], [36]. Accuracy is assessed by reviewing the fuzzy elements and their quantifications against domain knowledge [37], [38]. Completeness is verified to confirm that all aspects of the domain are covered, and consistency checks are performed to identify and resolve conflicting definitions [39]. Reasoners such as FuzzyDL are employed to check the ontology's logical consistency [40]. Finally, understandability is evaluated to ensure that the ontology is comprehensible to all stakeholders, including domain experts and intended users [41].

 ISSN 2723-6471

4. A Use Case of Developing a Generic Fuzzy Rough Context Ontology using FUZRUF-Onto

4.1. Phase 1: Ontology Purpose and Motivation (Establishing the Need for Fuzzy Rough Ontology)

Modern users rely on handheld devices to access data and request services based on context-related information like location, time, and environment. Context-aware information enhances usability by enabling applications to adapt dynamically. Often, users describe their needs with vague linguistic terms, such as "a close restaurant to my workplace." Addressing such vagueness requires Fuzzy Rough Description Logics, which define linguistic variables and modifiers to assign truth degrees based on certainty levels.

In context-aware domains, information modeling often revolves around the "5W questions" (Who, What, When, Where, and Why). Relationships in the ontology must represent user contexts, such as linking users, devices, and locations. Linguistic thresholds, defined by domain experts, are used to distinguish vague terms like close to or far from. These terms overlap and require fuzzification to meet domain-specific requirements. For example, stating "The park is very close to the restaurant" emphasizes a stronger degree of proximity than "close." This resemblance can be modeled as a fuzzy relation R on a set x, where $R(x, y)$ represents the degree to which x resembles y. Resemblance relations are defined using fuzzy necessity (intensifying) and fuzzy possibility (weakening) operators [28], [31], [33]. Definition (Resemblance relation). For a universe X, a relation R on X is a resemblance relation if for all (x, y) in X:

$$R(x,x) = 1, \; R(x,y) = R(y,x) \tag{4}$$

4.2. Phase 2: Determining the Required Ontology Elements

Domain experts and ontology engineers collaborate to clearly distinguish between fuzzy-related information and crisp information. This results in a knowledge base that is partitioned into two distinct parts: one containing precise information (crisp elements) and the other containing fuzzy and fuzzy rough-related information. The crisp elements are identified to represent the precise information required to model the targeted domain effectively. For clarity and to maintain focus within the scope of this publication, the ontology is limited to the following six main entities: User, Location, PersonalStatus, Environment, Activity, and Action. Together, these elements define the "5W" questions related to the user's context: Who, What, When, Where, and Why. Figure 5 illustrates the initial layout of the constructed ontology, showing the main concepts and their sub-concepts. This layout serves as a foundation for further development.

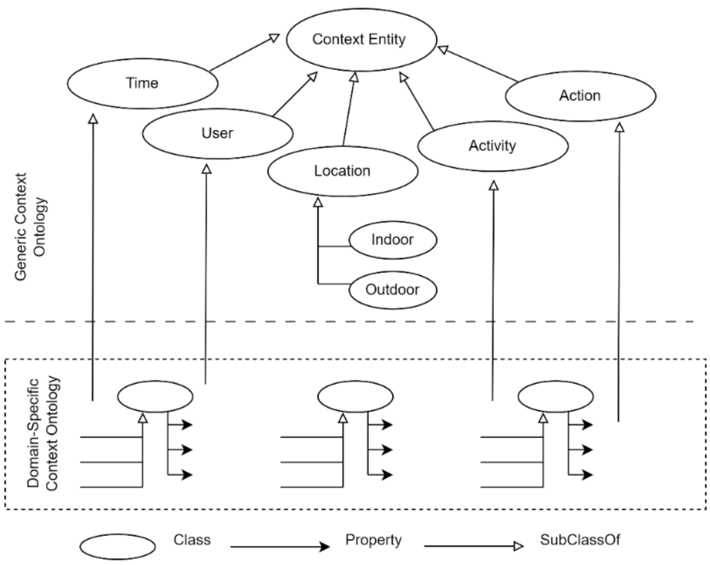

Figure 5. The Initial Layout of the Constructed Ontology

It is important to note that this initial ontology is designed as a generic context ontology. It can be extended with additional classes and properties to represent specific domain contexts, allowing for flexibility and adaptability in different use cases.

4.3. Phase 3: Reusing Existing Ontologies' Elements

In the constructed ontology, as shown in the concept's taxonomy, general concepts such as time and location can be reused from publicly available ontologies rather than being rebuilt from scratch. Leveraging existing ontologies ensures efficiency and consistency, while also enhancing interoperability across different systems [42], [43].

4.4. Phase 4: Defining The Ontology Elements

In Phase 2, the primary ontology entities that represent the user's context and their sub-concepts were identified. In this phase, these elements are further defined and structured.

The User class represents a unique individual with properties such as profile, calendar, and mobility status. This class is designed to observe users in their environment, with roles that depend on the domain, such as tourist, patient, lecturer, or shopper. The Action class defines atomic events performed by the user, each associated with a timestamp. Examples include EnterBuilding, OpenDoor, or TurnOnLight. The Activity class represents single or composite sets of actions with an inherent intent, along with StartDateTime and EndDateTime. Examples include AttendingMeeting or VisitingLocation. Some activities, such as Lecturing, can be divided into sub-activities like GiveLecture or AttendLecture. The Place class refers to locations in the environment where a user may be located, nearby, or far away. This can include indoor spaces, such as hospitals or homes, and outdoor spaces, such as parks or streets. The Time class provides a chronological representation of the context, defining the temporal aspects of the domain.

LocateAt relates a user to a specific place, while EngageIn links a user to an activity as a participant. HasCurrentTime associates a user with their current time, and Perform links a user to a specific action. Other relations include IsScheduledAt, which connects activities to allocated times, and IsOccurredAt, which connects actions to their corresponding times. Place properties include elements such as light and humidity. Time properties focus on start and end times. User properties include name, calendar, and personal status. After defining the ontology elements, ontology engineers and domain experts assess vagueness within the domain. While explicit vague concepts may not be present, vagueness can exist in relations and attributes, such as overlapping boundaries between terms.

Fuzzy datatypes represent vague attributes and are expressed using ranges defined with fuzzyDL syntax. For example, the User.hasMobility role specifies whether a user is still or moving at a certain speed. Similarly, User.hasDistance captures a user's proximity to a location, categorized as LocatedIn, CloseTo, or FarFrom.

Still and Moving represent user mobility. LocatedIn, CloseTo, and FarFrom capture spatial proximity. Cold, Warm, and Hot represent temperature. Table 2 summarizes these fuzzy datatypes, their definitions, and modeled vagueness.

Table 2. Fuzzy Datatypes Defined for Fuzzy Concrete Roles

Fuzzy Datatype	Definition	Vague Information Modeled
Still	Represents user mobility when stationary.	Left-shoulder membership function: left-shoulder (0, 100, 0, 3)
Moving	Represents user mobility when moving.	Right-shoulder membership function: right-shoulder (0, 100, 0.5, 3)
LocatedIn	Approximates the user's location within a place.	Trapezoidal membership function: Trapezoidal (0, 150, 0, 0, 50, 150)
CloseTo	Approximates proximity to a place.	Right-shoulder membership function: right-shoulder (0, 425, 50, 150)
FarFrom	Approximates distance away from a place.	Trapezoidal membership function: Trapezoidal (275, 1000, 275, 425, 725, 1000)
Cold	Represents cold temperatures.	Left-shoulder membership function: left-shoulder (0, 100, 0, 19)
Warm	Represents warm temperatures.	Right-shoulder membership function: right-shoulder (0, 100, 20, 25)
Hot	Represents high temperatures.	Left-shoulder membership function:

		left-shoulder (0, 100, 25, 70)

Fuzzy relations assign membership values to object properties in the ontology. IsScheduledAt connects activities to their timeframes, addressing borderline proximity to other activities. IsOccurredAt connects actions to their timeframes, accounting for potential overlap. Table 3 details the fuzzy datatypes for these relations.

Table 3. Fuzzy Datatypes Defined for Fuzzy Abstract Roles

Fuzzy Datatype	Definition	Vague Information Modeled
CloseToTime	Represents the degree of time proximity between activities or actions.	Left-shoulder membership function: left-shoulder (0, 100, 10, 20)
FarFromTime	Represents the degree of time separation between activities or actions.	Right-shoulder membership function: right-shoulder (0, 100, 15, 30)

Figure 6 illustrates the constructed context ontology, showing crisp and fuzzy classes, properties, and relations. Dashed rectangles represent generic concepts such as Time and Place, which are inherited from publicly available ontologies. Additional relations and properties may be added as required.

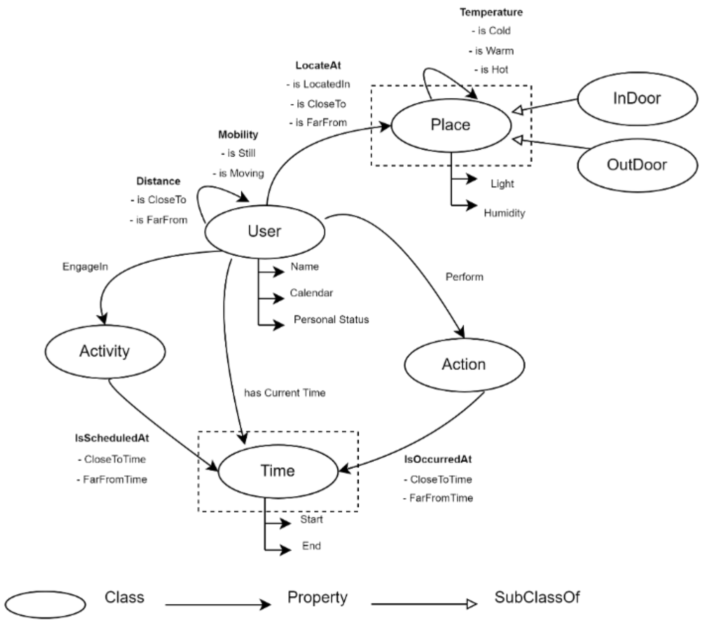

Figure 6. The Layout of the Constructed Ontology

4.5. Phase 5: Approximating the Vague Fuzzy Concepts/Sets Using Linguistic Modifiers/Hedges Through Fuzzy Rough Sets (Approximators)

Linguistic modifiers (also referred to as linguistic hedges) are specific types of linguistic expressions like very, extremely, more or less, quiet. While applied to adjectives, linguistic hedges allow us to express an emphasis we impose on the corresponding properties. They can be considered as special expressions by which the degree of membership of fuzzy datatype/relation could be modified. hedges such as very, extremely, definitely, more or less, roughly, almost, possible, etc., are utilized to describe the vague properties of fuzzy concepts. These modifiers are modeled by means of the construction of upper and lower approximations of the identified fuzzy concepts and relations. For example, given the fuzzy datatype property CloseTo between (User, Place), the fuzzy modifier Very is identified to modify the value of the membership function of that datatype and hence, the newly identified VeryCloseTo datatype makes the user not only CloseTo to a specific location, but VeryCloseTo to that location.

Linguistic fuzzy rough sets (also called fuzzy necessity and fuzzy possibility) will be used to tune the membership function values of four different fuzzy properties: CloseTo, Moving, Warm, and CloseToTime according to users' preferences and requirements. This could be achieved through using Eq. (1) and (2). For example, the fuzzy property "CloseTo" can be expressed by the resemblance relation

$$R(x, y) = \max\left(0, \min\left(1, 3.5 - \frac{|x-y|}{100}\right)\right) \tag{5}$$

This relation expresses the relation, specifically, the similarity between any two members x and y of the universe $X = [0, +\infty)$ of distance. The role of domain expert is to find the appropriate relation to represent this similarity.

$$\mu(\text{closeTo}) = \begin{cases} \frac{x-0}{500-0} & 0 \le x \le 500 \\ 1 & 500 < x \le 1000 \\ \frac{1500-x}{1500-1000} & 1000 < x \le 1500 \\ 0 & x > 1500, x < 0 \end{cases} \tag{6}$$

As shown from the defined membership function μ, a place that distant 600M is close to the user at a degree of 1 (the value of the μ). However, if the user decided to discover the places that are "Very" close, this value will be tuned for the same distance using Eq. (1), i.e., the lower approximation of the fuzzy term "CloseTo". The lower approximation of the fuzzy term "CloseTo" can be calculated to find the new membership value for the same place when it required to be considered as "VeryCloseTo". By finding the greatest lower bound inf of the set {1 − R(x; y); F(y)} => {.8,1,1,1,1,1,1,1,1,1,1,1,1,1}. The inf of the derived set is .8. So, a place that is 600M distant is VeryCloseTo to the user with .8 degree. Figure 7 illustrates the membership function values for the "CloseTo" and "VeryCloseTo" distances.

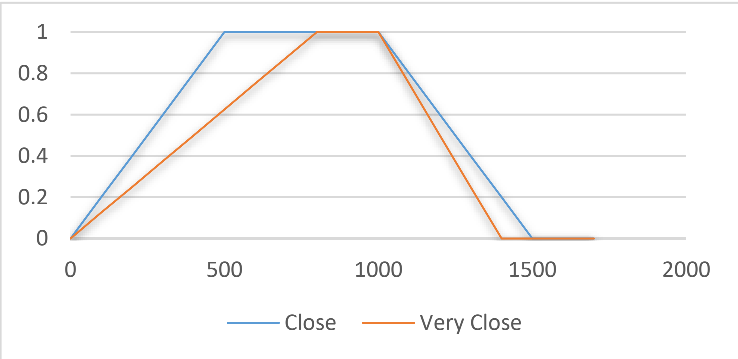

Figure 7. Membership Function Values for Close and VeryClose Distance

The second fuzzy property is "temperature" which is a property of the "Place" entity. The fuzzy property "Warm" that represents one of the temperature conditions can be expressed using the resemblance relation

$$R(x, y) = \min\left(1, \max\left(0, 2 - \frac{|x-y|}{2}\right)\right) \tag{7}$$

$$\mu(\text{Warm}) = \begin{cases} \frac{x-0}{19-0} & 0 \le x \le 19 \\ 1 & 19 < x \le 24 \\ \frac{28-x}{28-24} & 24 < x \le 28 \\ 0 & x > 28, x < 0 \end{cases} \tag{8}$$

As seen from the defined membership function μ, the temperature degree 14 is considered warm with .7 degree. However, the same temp. degree is considered "VeryWarm" with just .6 degree. It is worth noting that the linguistic modifier "Very" again used to tune this value with the lower approximation of the fuzzy term "Warm" using Eq. (1), i.e., the lower approximation of the fuzzy term "Warm". Figure 8 depicts the difference between the truth values of the term "Warm" and "VeryWarm" temperature.

ISSN 2723-6471

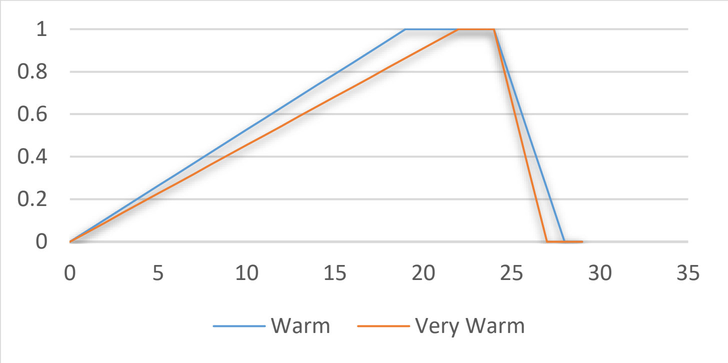

Figure 8. Membership Function Values for Warm and VeryWarm Temperature

The third fuzzy property in the constructed ontology is the "Mobility" by which the mobility of a specific user, throughout estimating his/her speed, is evaluated. The membership function illustrates the truth degrees of the user mobility according to the detected speed. The fuzzy property "Moving" that represents the user mobility can be expressed using the resemblance relation, and fuzzy membership function

$$R(x, y) = \text{Max}\,(0, \min\,(1\,,\, 1.1 - \frac{(x-y)^2}{10})) \tag{9}$$

$$\mu(\text{Moving}) = \begin{cases} 1 & x < 2 \\ \frac{5-x}{3} & 2 < x \leq 5 \\ 0 & x > 5 \end{cases} \tag{10}$$

For example, if the user is moving at 3 km/h speed, we can say that he is moving with .7 degree. However, in some cases, we need to discover the "Possibility" that this user is "Moving". In such case, the upper approximation of this fuzzy set needs to be calculated using Eq. (2). The upper approximation of the fuzzy term "Moving" can be calculated to find the new membership value for the user status when it required to be considered as "PossiblyMoving". By finding the least upper bound sup of the set {R(x; y); F(y)} => {.2,.7,1,.7,.4,.9,0,0}. The sup of the derived set is 1. So, a user with moving speed 3 is PossiblyMoving with 1 truth degree.

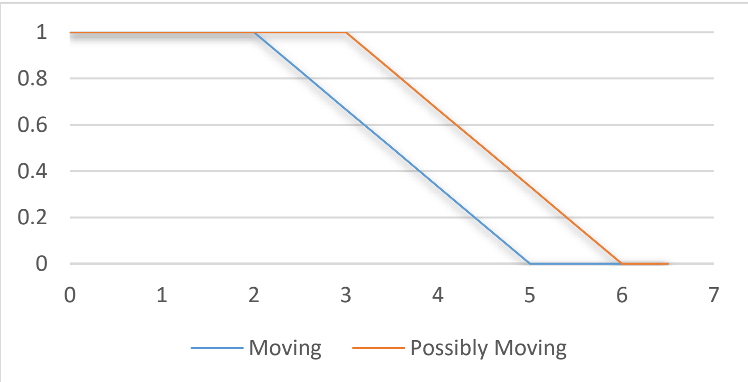

Figure 9. Membership Function Values for Still and PossiblyMoving Situation

Figure 9 depicts how values of the membership function of different speeds were changed when the "possibility" of user movement is considered. For instance, the truth value is increased from .5 to .8 when checking whether a user with 3.5 km/h speed is "PossiblyMoving" rather than just "Moving". The two fuzzy relations IsScheduledAt/ IsOccurredAt express the time approximation of a specific event/action to be started/occurred. The following is the membership function that can be used to calculate the truth value of the difference (in minutes) between two specific events, activities, actions or between user's current time and upcoming activities, or actions.

ISSN 2723-6471

$$\mu(\text{CloseToTime}) = \begin{cases} 1 & x < 30 \\ \frac{50-x}{20} & 30 < x \leq 50 \\ 0 & x > 50 \end{cases} \quad (11)$$

As shown, the 43 minutes difference between two activities makes them “CloseToTime” with .4 degree. On the other hand, this value is increased up to .6 when checking whether these two activities are “PossiblyCloseToTime” to each other using Eq. (2), i.e., the upper approximation of the fuzzy datatype “CloseToTime”. Figure 10 shows the difference between these membership values considering “CloseToTime” and “PossiblyCloseToTime” fuzzy datatypes.

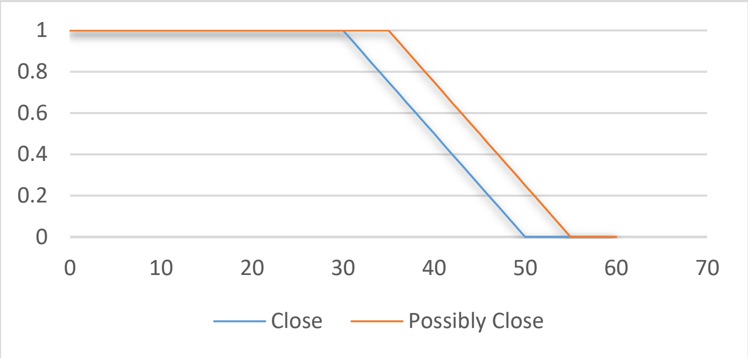

Figure 10. Membership Function Values for Close and PossiblyClose Activity/Action

4.6. Phase 6: Formalizing the Constructed Ontology

In this use case, OWL2 could be selected as the formalism language to represent the designed ontology model. Ontology editors such as the fuzzy-OWL protégé extension could be utilized in this phase. Protégé provides an easy user-friendly tool to visually implement the designed ontologies and allows for automatic generation of the code in different languages such as OWL and RDF. For example, Syntax and semantics of RDF are extended to support real numbers of the interval [0,1] to be expressed as degrees subjects, objects, and predicates [44]. In addition, there are a set of fuzzy extensions of description logics as in [45] that could be utilized to enable this transformation process. Bobillo et al. [37], and Nilavu and Sivakumar [4] introduced a concrete methodology to formalize fuzzy and fuzzy rough ontologies using OWL2 annotation properties. It is worth noting that different ontology formalism languages vary from each other in terms of characteristics, rules, and capabilities they have. There is no standard mechanism to evaluate these languages regarding their strength and weakness in representing a specific ontology element. Therefore, the formalism language should be chosen according to constructed ontology’s requirements.

4.7. Phase 7: Validating the Constructed Ontology

The validation results confirm the usefulness of the developed fuzzy rough ontology. While its consistency can be objectively evaluated using semantic reasoners such as the fuzzyDL reasoner, other features like correctness, accuracy, completeness, and understandability require subjective evaluation by ontology engineers, domain experts, and users involved in the development process. Since this ontology serves as a generic fuzzy rough context ontology, existing literature was used as a baseline to assess these features.

The correctness of the ontology is evident in its ability to serve various application scenarios as a generic reference, which can be adapted for different domains. The vagueness reflected in the fuzzy elements is accurately captured using well-defined fuzzy sets, and the boundary between crisp and fuzzy elements is clearly delineated, ensuring that the crisp elements contain no vague meanings. The correctness was validated by comparing the ontology’s compatibility with other context ontologies in the literature [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57].

In terms of accuracy, the ontology effectively quantifies the fuzzy elements and approximates their degrees of vagueness intuitively. These aspects were validated against comparable studies in the literature [32], [58], [59], [60], [61], and further reviewed by domain experts to ensure that the fuzzy degrees align with intuitive interpretations. The ontology is also complete, as it meets all the requirements for representing basic context concepts, their properties, individuals, and relationships. It successfully captures the vagueness identified during the development phases while addressing all knowledge requirements. The completeness of the ontology was confirmed by comparing it with existing works in the literature [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57].

Consistency was validated through the implementation of a simple scenario using the Fuzzy OWL2 language. This scenario involved deriving context information for a user, including their location, mobility status, personal status, upcoming events, and weather conditions. By adding instances and property values, the reasoning process conducted with the FuzzyDL reasoner [34] revealed no inconsistencies in the Knowledge Base (KB). Additionally, the ontology's structure and content exhibited no controversial definitions related to vagueness. Finally, the ontology is highly understandable due to the use of self-explanatory terms. This makes it accessible and easy to comprehend for domain experts, ontology engineers, and end-users, further enhancing its practical usability across different domains.

5. Discussion and conclusion

The proposed methodology provides a structured and logical approach to building fuzzy rough ontologies, offering a clear set of activities to guide the development process. Its primary objective is to deliver a methodological framework that ensures better performance compared to intuitive or ad-hoc approaches. As the first methodology to combine fuzzy set and rough set theories, FUZRUF-onto precisely identifies context-dependent characteristics of objects, enabling a more accurate representation of user preferences by tuning truth levels in inferred contexts. Demonstrated in the practical use case, the step-by-step guidelines of FUZRUF-onto ensure each phase has clear objectives and tasks, enhancing both efficiency and accuracy in ontology development. Despite challenges in performing quantitative comparisons with existing methodologies—such as METHONTOLOGY [63], NeON [64], or Diligent [65], which also lack formal evaluations—the FUZRUF-onto methodology stands out due to its detailed descriptions of vague knowledge and explicit interpretations of fuzzy degrees. Unlike traditional non-methodological approaches, it provides a well-structured framework, balancing conceptual knowledge representation with practical development. Its clear distinction between crisp and fuzzy knowledge enables ontology engineers to focus on modeling vagueness while reusing existing methodologies for crisp elements. Furthermore, the methodology supports the reuse and sharing of developed ontologies across domains by categorizing knowledge into reusable components. As the first comprehensive guide for constructing fuzzy rough ontologies from scratch, FUZRUF-onto allows developers to define, approximate, and validate fuzzy elements effectively. By employing linguistic fuzzy rough sets and modifiers such as "extremely," "very," and "roughly," the methodology fine-tunes membership values to align with user preferences. Overall, FUZRUF-onto sets a new standard for ontology development, offering a practical, systematic, and innovative approach to constructing accurate and adaptable ontologies.

6. Declarations

6.1. Author Contributions

Conceptualization: R.S. and M.A.; Methodology: R.S.; Software: R.S.; Validation: R.S. and M.A.; Formal Analysis: R.S. and M.A.; Investigation: R.S.; Resources: R.S; Data Curation: M.A.; Writing Original Draft Preparation: R.S. and M.A.; Writing Review and Editing: M.A. and R.S.; Visualization: R.S. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

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

References

[1] R. Sanyour, "FUZRUF-onto: a Methodology to Develop Fuzzy Rough Ontologies," *self-Preprint*, Feb. 2024.

[2] I. Jurisica, J. Mylopoulos, and E. Yu, "Ontologies for Knowledge Management: An Information Systems Perspective," *Knowl. Inf. Syst.*, vol. 6, no. 1, pp. 380–401, 2004.

[3] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: principles and methods-Source link," Data and Knowledge Engineering , vol. 25, no. 1-2, pp. 161-197, 1998.

[4] D. Nilavu and R. Sivakumar, "Knowledge Representation Using Type-2 Fuzzy Rough Ontologies in Ontology Web Language," *Fuzzy Inf. Eng.*, vol. 7, no. 1, pp. 73–99, 2015.

[5] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquis.*, vol. 5, no. 2, pp. 199–220, 1993.

[6] C. Thomas and A. Sheth, "On the Expressiveness of the Languages for the Semantic Web-Making a Case for 'A Little More,'" *Capturing Intell.*, vol. 1, no. 1, pp. 3–20, 2006.

[7] P. Alexopoulos, M. Wallace, K. Kafentzis, and D. Askounis, "Utilizing Imprecise Knowledge in Ontology-based CBR Systems through Fuzzy Algebra," *Int. J. Fuzzy Syst.*, vol. 12, no. 1, pp. 1-9, 2010.

[8] S. Calegari and E. Sanchez, "A fuzzy ontology-approach to improve semantic information retrieval," in *CEUR Workshop Proceedings*, vol. 327, no. 1, pp. 1-7, 2007.

[9] U. Straccia, *Foundations of Fuzzy Logic and Semantic Web Languages*, vol. 148. Chapman and Hall, 2013.

[10] M. De Cock, C. Cornelis, and E. E. Kerre, "Fuzzy rough sets: The forgotten step," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 1, pp. 121–130, 2007.

[11] A. Kaklauskas, "Intelligent Decision Support Systems," *Biometric Intell. Decis. Mak. Support*, 2015.

[12] H. Thiele, "Fuzzy Rough Sets versus Rough Fuzzy Sets An Interpretation and a Comparative Study using Concepts ofModal Logics," *5th Eur. Congr. Intell. Tech. Soft Com- puting (EUFIT'97), Aachen, Ger.*, no. April, 1998.

[13] X. Li, J. F. Martinez, and G. Rubio, "A New Fuzzy Ontology Development Methodology (FODM) Proposal," *IEEE Access*, vol. 4, no. 1, pp. 7111–7124, 2016.

[14] P. Alexopoulos, M. Wallace, K. Kafentzis, and D. Askounis, "IKARUS-Onto: A methodology to develop fuzzy ontologies from crisp ones," *Knowl. Inf. Syst.*, vol. 32, no. 3, pp. 667–695, 2012.

[15] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.

[16] P. Hajek, *Methamatematics of Fuzzy Logic*. Springer, 1998.

[17] G. J. Klir and B. Yuan, "Fuzzy sets and fuzzy logic : theory and applications," p. 574, 1995.

[18] Z. Pawlak, "Rough sets," *Int. J. Comput. Inf. Sci.*, vol. 11, no. 5, pp. 341–356, 1982.

[19] D. Dubois and H. Prade, "Rough fuzzy sets and fuzzy rough sets," *Int. J. Gen. Syst.*, vol. 17, no. 2–3, pp. 191–209, 1990.

[20] N. F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," *Sustain.*, vol. 9, no. 12, pp. 1–25, 2017.

[21] Z. R. Samani and M. Shamsfard, "The state of the art in developing fuzzy ontologies: A survey," *arXiv*, vol. 2018, no. 5, pp. 22–22, 2018.

[22] H. Ghorbel, A. Bahri, and R. Bouaziz, "Fuzzy ontologies model for semantic web," 2010, vol. 2010, no. 1, pp. 10–10.

[23] R. Y. K. Lau, D. Song, Y. Li, T. C. H. Cheung, and J. X. Hao, "Toward a fuzzy domain ontology extraction method for adaptive e-learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 6, pp. 800–813, Jun. 2009.

[24] Q. T. Tho, S. C. Hui, A. C. M. Fong, and T. H. Cao, "Automatic fuzzy ontology generation for semantic web," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 6, pp. 842–856, Jun. 2006.

[25] S. Calegari and D. Ciucci, "Fuzzy ontology and fuzzy-OWL in the KAON project," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, vol. 2007, no. 1, pp. 5–5, 2007.

[26] A. Jain and A. Sharma, "Membership function formulation methods for fuzzy logic systems: A comprehensive review," *Journal of Critical Reviews*, vol. 7, no. 19, pp. 1–10, 2020.

[27] H. Wang and Z. Xu, "Linguistic fuzzy rough sets for multi-criteria group decision making," in *Proceedings of the International Conference on System Engineering and Engineering Management (SEEM)*, vol. 2015, no. 1, pp. 5–5, 2015.

[28] M. De Cock and E. E. Kerre, "Fuzzy modifiers based on fuzzy relations," *Information Sciences*, vol. 160, no. 1, pp. 173–199, 2004.

[29] S. Calegari and F. Farina, "Fuzzy ontologies and scale-free network analysis," *International Journal of Computer Science Applications*, vol. 4, no. 2, pp. 125–144, 2007.

[30] C. Cornelis, M. De Cock, and A. M. Radzikowska, "Fuzzy rough sets: From theory into practice," in *Handbook of Granular Computing*, vol. 2008, no. 1, pp. 20–20, 2008.

[31] M. De Cock, A. M. Radzikowska, and E. E. Kerre, "A fuzzy-rough approach to the representation of linguistic hedges," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, vol. 2002, no. 1, pp. 10–10, 2002.

[32] D. Pei, "A generalized model of fuzzy rough sets," *International Journal of General Systems*, vol. 34, no. 5, pp. 603–613, 2005.

[33] A. M. Radzikowska, "Fuzzy modal operators and their applications," *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 11, no. 1, pp. 10–10, 2017.

[34] F. Bobillo and U. Straccia, "The fuzzy ontology reasoner fuzzyDL," *Knowledge-Based Systems*, vol. 95, no. 3, pp. 12–34, Mar. 2016.

[35] Z. Jun, S. Lixin, Z. Zhou, and L. Yan, "Fuzzy ontology model for knowledge management," *International Journal of Computational Intelligence Systems*, vol. 2007, no. 1, pp. 10–10, 2007.

[36] Z. Shi and D. Leake, "Fuzzy ontology models based on fuzzy linguistic variables for knowledge management and information retrieval," vol. 2008, no. 1, pp. 10–10, 2008.

[37] F. Bobillo and U. Straccia, "Fuzzy ontology representation using OWL 2," *International Journal of Approximate Reasoning*, vol. 52, no. 7, pp. 1073–1094, 2011.

[38] Y. Jiang, J. Wang, P. Deng, and S. Tang, "Reasoning within expressive fuzzy rough description logics," *Fuzzy Sets and Systems*, vol. 160, no. 23, pp. 3403–3424, 2009.

[39] F. Bobillo and U. Straccia, "Generalized fuzzy rough description logics," *Information Sciences*, vol. 189, no. 1, pp. 43–62, 2012.

[40] F. Bobillo, M. Delgado, and J. Gómez-Romero, "DELOREAN: A reasoner for fuzzy OWL 2," vol. 2008, no. 1, pp. 10–10, 2008.

[41] F. Hobbs and J. Pan, "Time ontology in OWL," *W3C Working Draft 27*, vol. 2006, no. 9, pp. 10–10, Sep. 2006. [Online]. Available: https://www.w3.org/TR/owl-time/. [Accessed: 04-Oct-2021].

[42] T. Ding, L. Chen, H. Kagal, and L. Finin, "Public address ontology. DARPA Agent Markup Language (DAML) Program, DAML Tools for supporting Intelligent Information Annotation, Sharing and Retrieval," vol. 2011, no. 1, pp. 10–10, 2011.

[43] U. Straccia, "A minimal deductive system for general fuzzy RDF," *International Conference on Web Reasoning and Rule Systems*, vol. 2009, no. 1, pp. 166–181, 2009.

[44] U. Straccia, "Managing uncertainty and vagueness in description logics, logic programs and description logic programs," *Reasoning Web. Lecture Notes in Computer Science*, vol. 2008, no. 1, pp. 54–103, 2008.

[45] Z. J. Lu, G. Y. Li, and Y. Pan, "A method of meta-context ontology modeling and uncertainty reasoning in SWoT," Proceedings of the 2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), vol. 2017, no. 1, pp. 128–135, 2017.

[46] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.

[47] K. L. Skillen, L. Chen, C. D. Nugent, M. P. Donnelly, W. Burns, and I. Solheim, "Ontological user profile modeling for context-aware application personalization," *Lecture Notes in Computer Science*, vol. 7656, pp. 261–268, 2012.

[48] X. H. Wang, D. Q. Zhang, T. Gu, and H. K. Pung, "Ontology based context modeling and reasoning using OWL: CONON, the context ontology," *Proceedings of the IEEE Annual Conference on Pervasive Computing and Communications Workshops (PerCom)*, vol. 2004, no. 1, pp. 10–10, 2004.

ISSN 2723-6471

[49] D. Riboni and C. Bettini, "OWL 2 modeling and reasoning with complex human activities," vol. 2009, no. 1, pp. 10–10, 2009.

[50] K. Mitra, A. Zaslavsky, and C. Ahlund, "Context-aware QoE modelling, measurement, and prediction in mobile computing systems," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 920–936, 2015.

[51] M. Aziez, S. Benharzallah, and H. Bennoui, "An ontology-based context model for the discovery of IoT services in the Internet of Things," *Proceedings of the International Conference on Mathematics and Information Technology (ICMIT)*, vol. 2018, no. 1, pp. 209–213, 2018.

[52] Y. Chen, J. Zhou, and M. Guo, "A context-aware search system for Internet of Things based on hierarchical context model," *Telecommunication Systems*, vol. 62, no. 1, pp. 77–91, 2016.

[53] D. Ejigu, M. Scuturici, and L. Brunie, "An ontology-based approach to context modeling and reasoning in pervasive computing," *Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, vol. 2007, no. 1, pp. 14–19, 2007.

[54] C. Bettini et al., "A survey of context modelling and reasoning techniques," *Pervasive and Mobile Computing*, vol. 6, no. 2, pp. 161–180, 2010.

[55] H. E. Degha, F. Z. Laallam, and B. Said, "Intelligent context-awareness system for energy efficiency in smart building based on ontology," *Sustainable Computing: Informatics and Systems*, vol. 21, no. 1, pp. 212–233, 2019.

[56] S. Sasirekha, S. Swamynathan, and S. Keerthana, "A generic context-aware service discovery architecture for IoT services," *Communications in Computer and Information Science*, vol. 808, pp. 273–283, 2018.

[57] J. Mok and H. Min, "Ontology-based context-aware model by applying Bayesian network," *Proceedings of the 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, vol. 2015, no. 1, pp. 2660–2664, 2016.

[58] P. Korpipää, J. Mäntyjärvi, J. Kela, H. Keränen, and E. J. Malm, "Managing context information in mobile devices," *IEEE Pervasive Computing*, vol. 2, no. 3, pp. 42–51, 2003.

[59] N. Díaz Rodríguez, M. P. Cuéllar, J. Lilius, and M. Delgado Calvo-Flores, "A fuzzy ontology for semantic modelling and recognition of human behaviour," *Knowledge-Based Systems*, vol. 66, no. 1, pp. 46–60, 2014.

[60] H. E. Byun and K. Cheverst, "Supporting proactive 'intelligent' behaviour: The problem of uncertainty," *Management*, vol. 2003, no. 1, pp. 10–10, 2003.

[61] M. G. C. A. Cimino, B. Lazzerini, F. Marcelloni, and A. Ciaramella, "An adaptive rule-based approach for managing situation-awareness," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10796–10811, 2012.

[62] M. Ferndndez, A. Gomez-Perez, and N. Juristo, "METHONTOLOGY: From ontological art towards ontological engineering," *Ontology Engineering*, vol. 1997, no. 1, pp. 10–10, 1997.

[63] A. Gómez-Pérez and M. Carmen Suárez-Figueroa, "NeOn methodology for building ontology networks: A scenario-based methodology," vol. 2009, no. 1, pp. 10–10, 2009.

[64] H. Sofia Pinto, C. Tempich, D. Vrandeči, S. Pinto, and Y. Sure, "The DILIGENT knowledge processes," *Journal of Knowledge Management*, vol. 2005, no. 1, pp. 10–10, 2005.

[65] R. N. Carvalho, K. B. Laskey, and P. C. G. da Costa, "Uncertainty modeling process for semantic technology," *PeerJ Computer Science*, vol. 2016, no. 8, pp. 1–36, 2016.