http://dx.doi.org/10.5455/sf.69169

Science Forum (Journal of Pure and Applied Sciences)

journal homepage: www.atbuscienceforum.com

Modelling Uncertainty in Ontology for The Semantic Web Using Vague Graph Theory

Amina Hassan Abubakar*, Armand Florentin Donfack Kana

Department of Computer Science, Ahmadu Bello University, Zaria Kaduna State, Nigeria

ABSTRACT

Ontology is viewed as an explicit representation of conceptualization. It consists of a set of concepts, their definitions and their interrelationships. Ontologies are used to unambiguously model an area of knowledge by "semantically" defining the concepts and their relationships in a given domain in a way that can be understood by both humans and machines. One of the major limitations of traditional ontology formalism is its lack of support for the representation of uncertainty and imprecision. As a result, they cannot handle incomplete or partial knowledge. Attempts have been made by other researchers to address this problem using algebraic methods, such as rough sets, soft sets, fuzzy sets, and probabilistic approaches. However, the conceptualization of the proposed solutions in those approaches is not always the same as how ontologies are actually implemented. Ontology languages [such as Resource Description Framework (RDF), RDF Schema (RDFS), and Web Ontology Language (OWL)] are used for expressing ontologies for Semantic Web (SW) in a crisp manner. This paper proposed a vague graph approach to formalize uncertainty in ontology for the SW. The paper extended RDF_{vg} and $RDFS_{vg}$, respectively, where both syntactic and semantic extensions of the languages are proposed. Similarly, a description of vague OWL was provided as OWL_{vg}.

ARTICLE INFO

Article history:
Received 11 October 2019
Received in revised form
19 October 2019
Accepted 20 October 2019
Published 21 February 2020
Available online 21 February 2020

KEYWORDS

Uncertainty
RDF_{vg}
$RDFS_{vg}$
OWL_{vg}

1. Introduction

Semantic Web (SW) technologies that use descriptive markup languages, such as the Resource Description Framework (RDF), RDF Schema (RDFS), and Web Ontology Language (OWL), were proposed to model web content in a machine-readable way, so as to assist in information gathering and automatic search processing by software agents, thus improving cooperation between humans and computers. The SW, which has ontology as its core can be viewed as a web of data comparable to a globally accessible database. Ontological knowledge representation will provide the needed solution for achieving a successful semantic information representation (Zhongli et al., 2006).

Ontologies are seen as explicit representation of a conceptualization. The conceptualization includes a set of concepts, their definitions and the interrelationships among them. Ontology languages used for the semantic web, like RDF, RDFS and OWL are used to model knowledge about an application domain in a crisp manner. But in almost all aspect of ontology engineering, uncertainty exists. Therefore, there is the need to provide a formal syntax and semantics for ontology languages and ontological operations that take uncertainty into consideration, as it is very important in order to achieve interoperability between ontologies. This requires formalizing ontology languages and ontological operations at the semantic web layer. Most

* **Corresponding author** Amina Hassan Abubakar ✉ aminatgusau@yahoo.com ✉ Department of Computer Science, Ahmadu Bello University, Zaria Kaduna State, Nigeria.

ontologies are designed containing concepts and relations that describe stated realities about a given real world domain.

Uncertainty is an inherent feature of many tasks on the web. Thus, formalisms that are capable of representing and reasoning under uncertainty are needed for full realization of the World Wide Web as a source of processable data and services. According to Kenneth et al., (2008) even though semantic markup languages such as OWL can be used to represent qualitative and quantitative information about uncertainty, there are essentially no formal foundations established for doing that. Consequently, each developer is left with the task of forming his/her own set of constructs for uncertainty representation. However, this is a major obligation in an environment that depends on interoperability among systems and applications. Thus, there is the need to model uncertainty in ontologies, as well as to define an approximate inference mechanism that will enable ontologies reasoning under uncertainty which is lacking in the traditional ontologies. Semantic web languages are used to represent knowledge about established facts existing in the domain being modelled in a crisp manner only. However, it is a known fact that the real-world domains, which these ontologies are expected to model, are characterized by uncertainty; therefore, those uncertainties have to be considered in order to correctly model such domains. This lack of representation of uncertain information discourages existing RDF(S) and OWL users from reasoning over such uncertain knowledge. Extension languages are actually needed on RDF, RDFS and OWL to address this limitation. For example, we cannot express and / or reason over the statement 'Zainab is very pretty', because the degree to which she is pretty, is not expressible in RDF(S) or OWL even though these languages provide a mean of representing "pretty" in a crisp manner.

The aforementioned limitation necessitates extending web ontology languages and operations to make possible representation and consequently reasoning with uncertainty. In order to address the issue identified and presented, the theory of vague graph was employed in this research to illustrate the possibility of extending the RDF to vague RDF (RDF_{vg}) to account for uncertainty. Therefore, this paper presents a vague graph-based approach to formalize vague RDF (RDF_{vg}) and some ontology operations on the proposed RDF_{vg}

2. Vague Graph

This section reviews the fundamental concepts of vague graph needed in this paper. To arrive at this, concepts of vague graph obtained from Ramakrishna (2009) are presented in this section.

Definition 2.1: A vague set A on set X is a pair (t_A, f_A) where $t_A : X \to [0,1]$, *and* $f_A : X \to [0,1]$, such that $0 < t_A(x) + f_A(x) \leq 1$ for all $x \in X$. The mappings $t_A : X \to [0,1]$ *and* $f_A : X \to [0,1]$ are the degrees of true membership and false membership functions, respectively, for the element $x \in X$, such that the functions t_A and f_A satisfy the condition: $t_A \leq 1 - f_A$. It should be noted that vague set represents the membership values of vertices/edges in a vague graph.

The function $t_A(x)$ is considered as the lower bound for the degree of membership of x in A, whereas $f_A(x)$ is the lower bound for the degree of non-membership of x in A. Hence, the degree of membership of x in the vague graph set A is characterized by an interval: $[t_A(x),\ 1 - f_A(x)]$.

Definition 2.2: A crisp graph G consists of a non-empty finite set of elements V called the vertices or nodes, and E, a finite set of distinct unordered pairs of elements V called the edges.

Definition 2.3: Let X and Y be any non-empty sets, a vague relation from X to Y is defined as a vague set of $X \times Y$, denoted by $\mu = (t_\mu, f_\mu)$ where $t_\mu : X \times Y \to [0,1]$, $f_\mu : X \times Y \to [0,1]$ which satisfies the condition $t_\mu(x,y) + f_\mu(x,y) \leq 1, \forall (x,y) \in X \times Y$.

A vague relation μ on a set V is a vague relation from V to V. If σ is a vague set on a set V, then a vague relation μ on σ is a vague relation on V, that satisfies
$t_\mu(x,y) \leq \min\{t_\sigma(x), t_\sigma(y)\}$,
$f_\mu(x,y) \geq \max\{f_\sigma(x), f_\sigma(y)\}, \forall (x,y) \in V$.

Definition 2.4: Let V be a non-empty set, members of V are called vertices. A vague graph $G = (\sigma, \mu)$ with a set of vertices V is a pair of functions σ, μ, where σ is a vague set of V and μ is a vague relation on σ.

For simplicity and convenience purpose, the ordered pair (μ,ν) used to describe μ is denoted by $\mu\nu$.

Example 2.1: Let $V = \{v_1, v_2, v_3, v_4, v_5\}$ be a set of vertices of a graph G. A graph $G' = (\sigma, \mu)$ shown in Figure 1 is a vague graph, where $\sigma = (t_\sigma, f_\sigma)$ is a vague set on V given as:

$\sigma = \left\{ \begin{array}{l} (v_1, 0.5, 0.4), (v_2, 0.3, 0.6), (v_3, 0.3, 0.5), \\ (v_4, 0.3, 0.4), (v_5, 0.5, 0.3) \end{array} \right\}$ while $\mu = (t_\mu, f_\mu)$ is a vague relation on V given as

$$\mu = \begin{Bmatrix} (v_1v_2, 0.2, 0.8), (v_2v_3, 0.2, 0.7), (v_3v_4, 0.2, 0.6), \\ (v_4v_5, 0.3, 0.6), (v_5v_1, 0.4, 0.5), (v_2v_4, 0.2, 0.6), \\ (v_2v_5, 0.2, 0.7), (v_3v_5, 0.3, 0.6), (v_1v_4, 0.3, 0.5), \\ (v_1v_3, 0.3, 0.6) \end{Bmatrix}$$

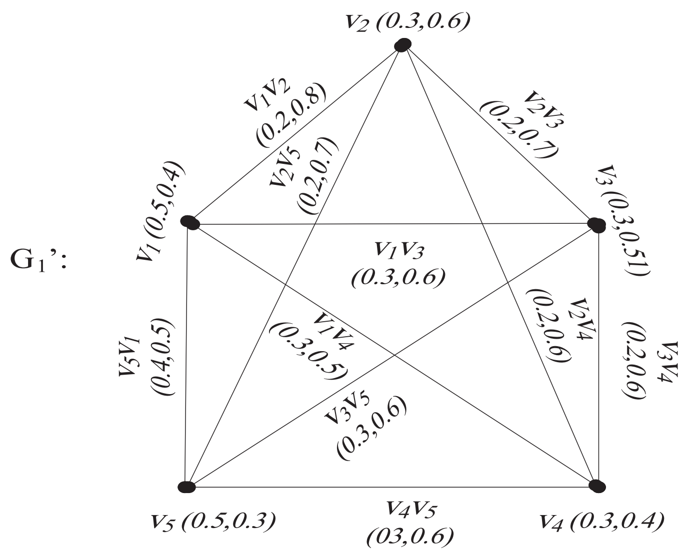

Figure 1. Vague graph G1′.

2. 1. Operations on Vague Graph

This subsection presents some operations on vague graph, such as union, intersection, and join, as defined in Ramakrishna (2009).

Definition 2.5: Union: Let $G_1' = (\sigma_1, \mu_1)$ and $G_2' = (\sigma_2, \mu_2)$ be two vague graphs of two crisp graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, respectively. Let the union of G_1 and G_2 be a crisp graph $G_3 = G_1 \cup G_2$. Then, the union of the two vague graphs G_1 and G_2 is a vague graph $G_3' = (\sigma_1 \cup \sigma_2, \mu_1 \cup \mu_2)$ defined as

$$V_{\sigma_1 \cup \sigma_2}(v) = \begin{cases} V_{\sigma_1}(v), & \text{if } v \in V_1 - V_2 \\ V_{\sigma_2}(v), & \text{if } v \in V_2 - V_1 \\ \max\{V_{\sigma_1}(v), V_{\sigma_2}(v)\} & \text{if } v \in V_1 \cap V_2 \end{cases}$$

$$V_{\mu_1 \cup \mu_2}(uv) = \begin{cases} V_{\mu_1}(uv), & \text{if } uv \in E_1 - E_2 \text{, i.e.,} \\ V_{\mu_2}(uv), & \text{if } uv \in E_2 - E_1 \\ \min\{V_{\mu_1}(uv), V_{\mu_2}(uv)\} & \text{if } uv \in E_1 \cap E_2 \end{cases}$$

$$t_{\sigma_1 \cup \sigma_2}(v) = \begin{cases} t_{\sigma_1}(v), & \text{if } v \in V_1 - V_2 \\ t_{\sigma_2}(v), & \text{if } v \in V_2 - V_1 \\ \max\{t_{\sigma_1}(v), t_{\sigma_2}(v)\} & \text{if } v \in V_1 \cap V_2 \end{cases}$$

$$f_{\sigma_1 \cup \sigma_2}(v) = \begin{cases} f_{\sigma_1}(v), & \text{if } v \in V_1 - V_2 \\ f_{\sigma_2}(v), & \text{if } v \in V_2 - V_1 \\ \max\{f_{\sigma_1}(v), f_{\sigma_2}(v)\} & \text{if } v \in V_1 \cap V_2 \end{cases}$$

$$t_{\mu_1 \cup \mu_2}(uv) = \begin{cases} t_{\mu_1}(uv), & \text{if } uv \in E_1 - E_2 \\ t_{\mu_2}(uv), & \text{if } uv \in E_2 - E_1 \\ \min\{t_{\mu_1}(uv), t_{\mu_2}(uv)\} & \text{if } uv \in E_1 \cap E_2 \end{cases}$$

$$f_{\mu_1 \cup \mu_2}(uv) = \begin{cases} f_{\mu_1}(uv), & \text{if } uv \in E_1 - E_2 \\ f_{\mu_2}(uv), & \text{if } uv \in E_2 - E_1 \\ \min\{f_{\mu_1}(uv), f_{\mu_2}(uv)\} & \text{if } uv \in E_1 \cap E_2 \end{cases}$$

Example 2.2: Let $V_1 = \{v_1, v_2, v_3, v_4\}$ and $V_1' = \{v_1, v_2, v_6\}$ be a set of vertices of two graphs G_1 and G_2, respectively. Figures 2 and 3 show the graphs $G_1' = (\sigma_1, \mu_1)$ and $G_2' = (\sigma_2, \mu_2)$, respectively. Where $\sigma_1 = (t_{\sigma_1}, f_{\sigma_1})$ is a vague set on V given as $\sigma_1 = \{(v_1, 0.5, 0.4), (v_2, 0.6, 0.2), (v_3, 0.3, 0.5), (v_4, 0.3, 0.4)\}$ while $\mu_1 = (t_{\mu_1}, f_{\mu_1})$ is a vague relation on V_1 given as $\mu_1 = \begin{Bmatrix} (v_1v_2, 0.4, 0.5), (v_1v_3, 0.2, 0.7), \\ (v_2v_4, 0.2, 0.6), (v_3v_4, 0.3, 0.4), \\ (v_2v_3, 0.2, 0.6), (v_4v_1, 0.4, 0.4) \end{Bmatrix}$, σ_2 is a vague set on V_2' given as $\sigma_2 = \{(v_1, 0.3, 0.1), (v_2, 0.5, 0.5), (v_6, 0.4, 0.1)\}$, and μ_2 is a vague relation on σ_2 given as $\mu_2 = \{(v_1v_2, 0.2, 0.6), (v_2v_6, 0.3, 0.6), (v_6v_1, 0.3, 0.3)\}$.

The union of G_1' and G_2' is a vague graph G_3' obtained as

$$\sigma_1 \cup \sigma_2 = \begin{Bmatrix} (v_1, 0.5, 0.4), (v_2, 0.5, 0.2), \\ (v_3, 0.3, 0.5), (v_4, 0.4, 0.3), (v_6, 0.4, 0.1) \end{Bmatrix}$$

$$\mu_1 \cup \mu_2 = \begin{Bmatrix} (v_1v_2, 0.4, 0.5), (v_2v_4, 0.2, 0.6), (v_3v_4, 0.3, 0.4), \\ (v_1v_3, 0.2, 0.7), (v_2v_3, 0.2, 0.6), (v_4v_1, 0.4, 0.4), \\ v_2v_6, (0.3, 0.6), v_6v_1, (0.3, 0.3) \end{Bmatrix}$$

Figure 4 shows a Vague graph G_3.

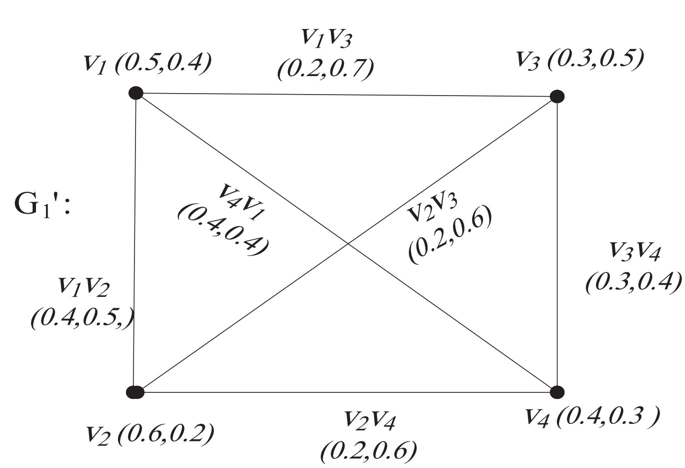

Figure 2. Vague graph G1′.

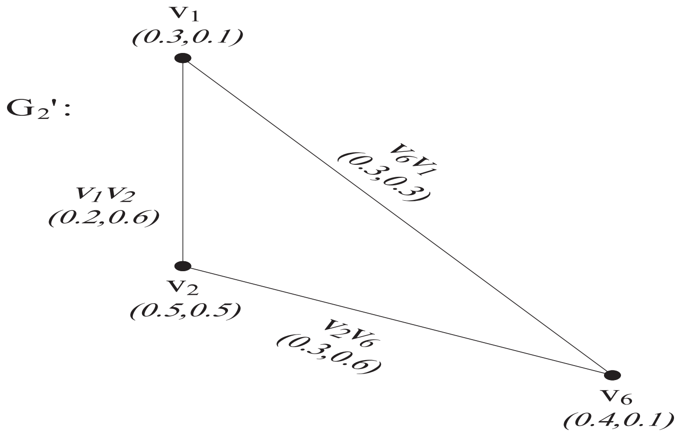

Figure 3. Vague graph G2'.

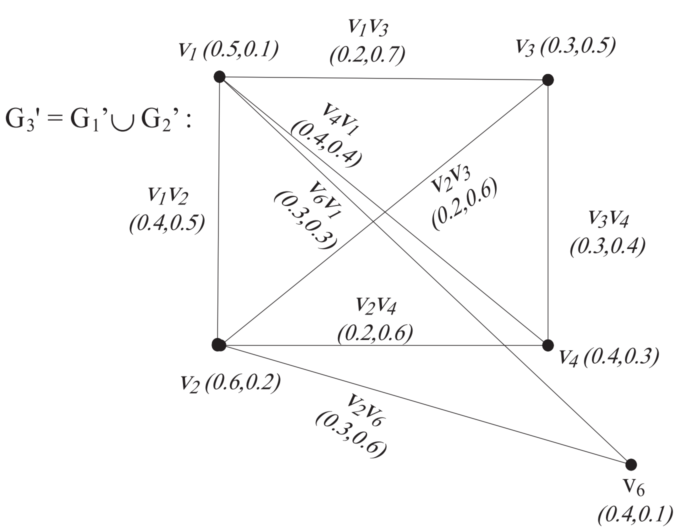

Figure 4. Union of two vague graphs G1' and G2.

3. Methodology

The vague graph-based model of ontology of uncertainty is defined to describe ontology languages in an uncertain manner by labeling objects and their relationship as nodes edges of a vague graph. The vague objects (nodes/vertices) represent things or resources that describe the subjects and objects of a statement while the arrows (arcs) represent functional relationships that describe the predicate of a statement.

The following considerations were made on ontological entities in order to use the vague graph to model uncertainty in ontologies into the semantic web ontological languages:

Objects: There are two types of objects represented in our proposed vague graph models: objects representing instances of a class and objects representing classes (the latter can be either crisp or vague). This research assumed that each of these objects corresponds to the nodes/vertices of the vague graph.

Instances: Since RDF is a data model that relates instance data, we assumed that vague RDF is also a data model that relates instance data in a way such that the membership of the instance data as well as the relationship is also represented.

Relations (Properties): These are the functional mappings from a class object or instance object or to the other. This represents the direct causal or functional relation from one object to another. Relations can be crisp or vague. This research considered relations (Properties) to be represented in both vague RDF and vague RDFS as edges of a vague graph with membership values.

Class: These are the set of things found in the domain of interest. Since classes and relationships existing between the classes are normally defined in RDFS, the same approach will be adopted to describe classes and the relationship that exists between classes in vague RDFS. Classes are represented as nodes with membership values.

3.1. Modeling semantic web languages using vague graph theory

In this subsection, this study proposed a unified semantics (i.e., the RDF-based semantics) for all the SW languages by extending them to represent connected vague graph structures. That is, the extended models of RDF_{vg}, $RDFS_{vg}$, and OWL_{vg} were proposed as extensions to RDF, RDFS, and OWL so that the anticipated connection between the languages is maintained while at the same time accounting for uncertainty at the semantic web stack.

Definition 3.1: A vague ontology system is defined as a pair of two sets $O_{vg} = (R, P)$, where R represents a set of resources *(e.g., concepts/classes* or *objects/ instances)* while P represents set of properties/attributes (*relationship*) provided at least an element of R or P is vague.

From definition 2.4, R is represented as $\sigma = (t_R, f_R)$ to denote the membership of the resources such that $t_R : R \to [0,1]$ and $f_R : R \to [0,1]$, and as $\mu = (t_P, f_P)$ to denote the membership of the properties such that $t_p : P \to [0,1]$ and $f_p : P \to [0,1]$.

Example 3.1: Let $R = \{r_1, r_2, r_3, r_4, r_5\}$ be the set of resources in O_{vg}, while $P = \{r_1r_2, r_1r_3, r_3r_4, r_2r_5, r_4r_5\}$ be the set of properties. The ontology $O_{vg} = (R, P)$, shown in Figure 5, is a vague ontology, where $\sigma = (t_R, f_R)$ is a vague set on resources R representing its membership which is given as: $\sigma = \{r_1, (0.7, 0.3), r_2, (1.0, 0.0), r_3, (0.0, 1.0), r_4, (0.6, 0.4) r_5, (0.8, 0.2)\}$ while $\mu = (t_P, f_P)$ is a vague relation on R representing its membership which is also given as:

$$\mu = \left\{ \begin{matrix} (r_1r_2, 0.2, 0.8), (r_1v_3, 0.7, 0.3), (r_3r_4, 0.4, 0.6), \\ (r_2r_5, 1.0, 0.0), (r_4v_5, 1.0, 0.0) \end{matrix} \right\}.$$

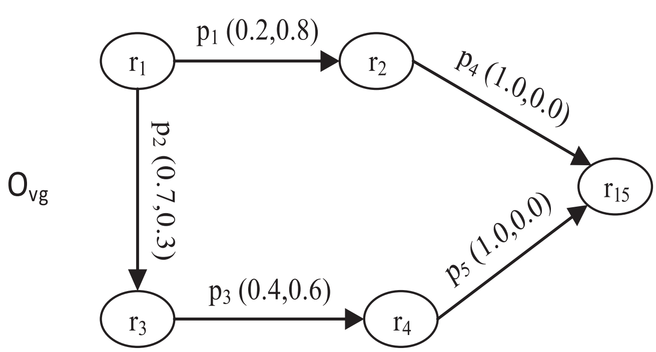

Figure 5. Ontology representation in a vague graph form.

We proposed some extensions to the existing SW Languages: RDF, RDFS, and OWL as RDF_{vg}, $RDFS_{vg}$, and OWL_{vg}, respectively, in order to enable the representation of vague knowledge in a domain of interest.

Three vague graph structures are used to define the three (3) extended languages as illustrated in Figure 6. The semantics of the proposed RDF_{vg}, $RDFS_{vg}$, and OWL_{vg} are based on RDF semantics. The RDF layer is modified to represent the vague graph model of RDF (RDF_{vg}), while the vague model of RDFS is shown in the $RDFS_{vg}$ layer and, finally, the model of vague OWL(OWL_{vg}) is obtained by allowing some properties on the objects defined in the OWL_{vg} layer.

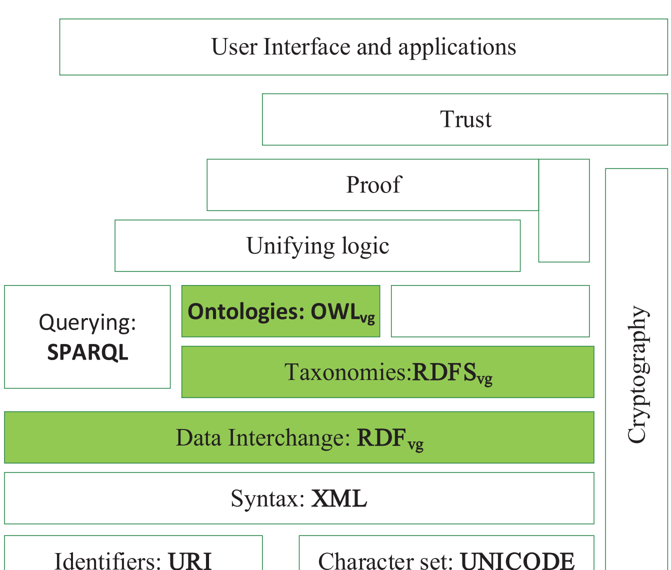

Figure 6. Modified layering architecture of the semantic web languages.

Modifications were only proposed at the RDF, RDFS, and OWL layers of the existing architecture. In place of the RDF layer, we proposed a RDF_{vg} layer that models RDF while taking uncertainty into account, the Uniform Resource Identifier/Internationalized Resource Identifier (URI/IRI) references are interpreted as objects (classes or individuals) and properties while the URI/IRI references in $RDFS_{vg}$ layer are interpreted as ontology classes, individuals, and properties. $RDFS_{vg}$ is an extension of RDFS. Furthermore, RDF_{vg} will provide equivalent support to the one provided by RDF in relating individuals to each other in the form of RDF triples. In the case of $RDFS_{vg}$, the mappings consider and represent the functional relationships between classes as well as individuals, for example, f (x) = y, where x and y represent classes/individuals with URI/IRI references. The consequence of allowing some properties in $RDFS_{vg}$ (e.g., Boolean combination of classes, subclassing, and inverse role) will produce OWL_{vg}.

In order to describe the vague graph-based syntax of RDF_{vg}, we made the following proposition to justify our rationale for the defined syntax.

Proposition 3.1: For every RDF graph, there is an equivalent vague RDF representation (RDF_{vg}) consisting of a set of vertices and a vague edge with their associated membership values.

Proof:

Let $R = \{r_1, r_2, r_3, \ldots r_n\}$ and $P = \{p_1, p_2, p_3, \ldots p_n\}$ be sets of resources and properties of an RDF statement, respectively. Let $R' = \{r_1', r_2', r_3', \ldots r_n'\}$ and $P' = \{p_1', p_2', p_3', \ldots p_n'\}$ be sets of vague resources and vague properties, respectively. An RDF statement in the form (s, p, o), where $s\, and\, o \in R$ and $p \in P$, can be represented in the form of RDF_{vg} graph as a pair (R', P') where R' denotes the vague set of $s\, and\, o$ and P denotes the vague set of p. That is, every RDF statement of the form (r_i, p_i, r_k), where $r_i, r_k \in R\, and\, p_i \in P$, has an equivalent vague RDF representation as $RDF_{vg} = (R', P')$, where R' denotes a vague set of $r_i'\ and\ r_k'\ \forall r_i', r_k' \in R'$ while P' denotes a vague set of $p_i', \forall p_i' \in P'$. Figures 7 and 8 show an RDF graph and its equivalent RDF_{vg}, respectively, where α, β represent the membership values.

Figure 7. An RDF Graph.

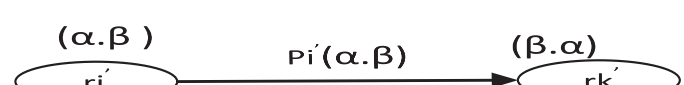

Figure 8. A vague RDF Graph.

Two functions $f:R\to R'$ and $g:P\to P'$ were constructed as

$f(r_i)=r_i'$

$f(r_k)=r_k'$

while

$g(p_i)=p_i'$

Table 1. Functions $f:R\to R'$ and $g:P\to P'$.

$f:R\to R'$		$g:P\to P'$	
R	R'	P	P'
r_i	r_i'	p_i	p_i'
r_k	r_k'		

The RDF graph and vague RDF graph shown in Figures 7 and 8, respectively, are *equivalent* since they have the same sets of vertices and edges, both have no loop, no parallel edges, 1 degree of vertex, only one component, and a pair of connected vertices, and both are fully connected. That is, everything is preserved and each of the graphs can be relabeled to the other QED.

3.2. Vague Graph-based RDF (RDF$_{vg}$)

This section presents the proposed extension of the standard RDF to vague RDF.

3.2.1. Syntactic model of vague RDF (RDF$_{vg}$)

The RDF syntax was extended to define a Vague Graph-Based RDF (RDF$_{vg}$) syntax as a pair (R,P) where R denotes the *Subjects / Objects* and P denotes the predicate as in the RDF triples. A formal definition of the vague RDF syntax is provided in definition 3.2 as

Definition 3.2: A vague RDF (RDF_{vg}) is a pair (R,P), where R represents a set of resources (representing a set of *Classes/Instances*) while P represents a set of properties/attributes (*relationship*) provided at least an element of $R \text{ or } P$ is vague. From definition 2.4, R is represented as $\sigma=(t_R,f_R)$, denoting a finite vague set on R such that $t_R:R\to[0,1]$ and $f_R:R\to[0,1]$, while P is represented as $\mu=(t_P,f_P)$, denoting a finite set of vague properties on R such that $t_p:P\to[0,1]$ and $f_p:P\to[0,1]$.

Example 3.2: Assuming that the property "*hasPrettyChild*" has vague values as α_1,β_1 where $\alpha_1=0.6$ and $\beta_1=0.4$, while the individuals "*Umar and Sumayya*" have vague values as α_2,β_2, where $\alpha_2=1.0$ and $\beta_2=0.0$ represent the true and false membership of the property and the individuals, respectively. A vague statement "*Umar* has a pretty child named *Sumayya*" can be represented in RDF_{vg} as shown in Figure 9.

Figure 9. RDF$_{vg}$ statement.

Example 3.3: Assuming that the property "*Knows*" has vague values as "1.0, 0.0," "*hasChild*" has "1.0, 0.0," "*hasPrettyChild*" has "0.6, 0.4," and "*isTallerThan*" *has* "0.7, 0.3," while the individuals "*Maryam, Zahra, Yasmin, Umar, and Sumayya*" has vague values as α_2,β_2, where $\alpha_2=1.0$ and $\beta_2=0.0$ represent the true and false membership of those properties and individuals, respectively. The statements *"Maryam has a mother named Zahra and a child named Umar," "Zahra knows Yasmin," "Umar has a pretty child named Sumayya,"* and "*Sumayya is taller than Yasmin"* can be represented in *RDF$_{vg}$* as shown in Figure 10.

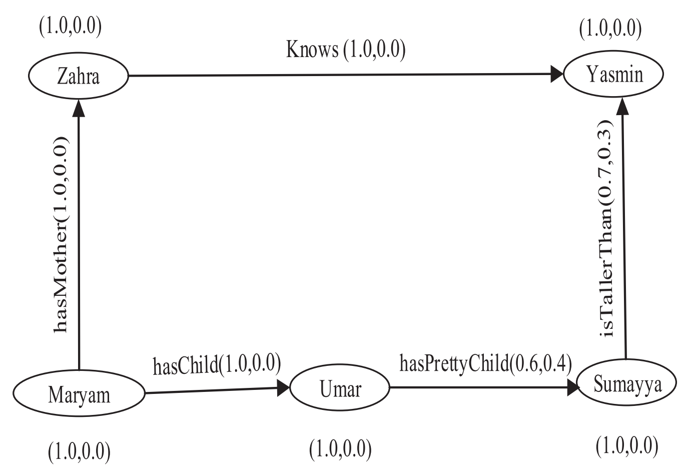

Figure 10. RDF$_{vg}$ statements.

$R=\{Zahra, Maryam, umar, Sumayya, Yasmin\}$, that is, $R=I=\{I1, I2, I3, I4, I5\}$, while

$$P=\begin{Bmatrix}\text{Knows(1.0,0.0), hasMother(1.0,0.0),}\\ \text{hasChild(1.0,0.0), hasPrettyChild(0.6,0.4),}\\ \text{isTallerThan(0.7,0.3)}\end{Bmatrix}=\{p1,p2,p3,p4,p5\}$$

3.2.2. Semantic model of vague RDF

The interpretation of objects (resources) and relationship (properties) for each vague graph structure representing RDF$_{vg}$ is based on modeling the RDF Interpretation. Similarly, each vague graph object is related to its interpretation via an interpretation function *I*.

To describe how *resources* (crisp and vague) are interrelated through properties (crisp and vague), triples were employed. Therefore, an interpretation comprises of (i) two sets *IR* (abstract resources) and

IP (properties) that may or may not be vague and (ii) a function IEXT that states which resources are interrelated by which properties.

This work adopted the semantics presented in Hayes and Patel-Schneider, (2014) which begins with an easy definition of simple interpretations of graphs, after that some additional criteria were provided in order to make the interpretations to be eligible as RDF interpretations. Lastly, additional constraints to be fulfilled by RDF interpretation so that it will be acknowledged as an RDFS-interpretation were provided.

The extension made on the adopted interpretation is as follows: the set of crisp resources are represented as IR_c while the set of vague resources are represented as IR_v, hence the set of all resources *IR* represent both crisp and vague, that is, $IR = IR_c \cup IR_v$. Similarly, the set of crisp properties are represented as IP_c while the set of vague properties are represented as IP_v; therefore, the set of all properties *IP* represents both crisp and vague properties, that is, $IP = IP_c \cup IP_v$, while other conditions are left the same.

3.3. Vague graph-based RDF Schema

The formalism of Vague RDF Schema ($RDFS_{vg}$) follows the same guidelines as the RDF_{vg}. The semantics of $RDFS_{vg}$ are stated in terms of "class." A class is a resource representing a set of things in the domain of discourse. Therefore, two classes can be related to each other via a subclass relationship (e.g., subClassOf), and similarly two properties can relate to one another via a subProperty relationship (e.g., subPropertyOf).

A formal definition of the $RDFS_{vg}$ is given in this section where it is defined in terms of vague classes and properties as follows:

Classes: The elements of *R* in $RDFS_{vg}$ represent vague class names. Vague relationship between classes R1 and R2 is represented as the mapping from R1 to R2, where $R1, R2 \in R$.

SubClasses: Let R1 and R2 denote two classes in $RDFS_{vg}$, where R1 is a subclass of R2 *iff* R1 is contained in R2.

Properties: In $RDFS_{vg}$, all elements of *P* represent properties (crisp/vague) which denote that a relationship exists between the objects (classes/instances). These relations include *instanceOf, domainOf, rangeOf.*

SubProperties: Let *P1* and *P2* be two properties (crisp/vague) in $RDFS_{vg}$, where *P1* is a *subproperty* of P2, *iff P1* is a *subrelation* of *P2*.

Definition 3.3: A vague RDF Schema ($RDFS_{vg}$) is a pair (R, P), where *R* represents set of resources (Classes) while *P* represents finite set of properties/ attributes (*relationship*), provided that, at least an element of *R* or *P* is vague. From definition 2.4, *R* is represented as $\sigma = (t_R, f_R)$, denoting finite vague set on R such that $t_R : R \to [0,1]$ *and* $f_R : R \to [0,1]$, while *P* is represented as $\mu = (t_P, f_P)$, denoting finite set of vague properties on *R* such that $t_p : P \to [0,1]$ *and* $f_p : P \to [0,1]$.

Example 3.4: Assuming the statement "*GoodUniverity and BestUniveristy are subClassOf a University.*" Let the concepts *"GoodUniversity," "BestUniversity,"* and

"University" have vague values α_1, β_1, α_2, β_2, and α_3, β_3, respectively. Where $\alpha_1 = 1.0$, $\beta_1 = 0.0$, $\alpha_2 = 0.6$, $\beta_2 = 0.4$, $\alpha_3 = 0.8$, and $\beta_3 = 0.2$, while the property *subClassOf* has vague α_1, β_1. The statement can be represented in $RDFS_{vg}$ as shown in Figure 11.

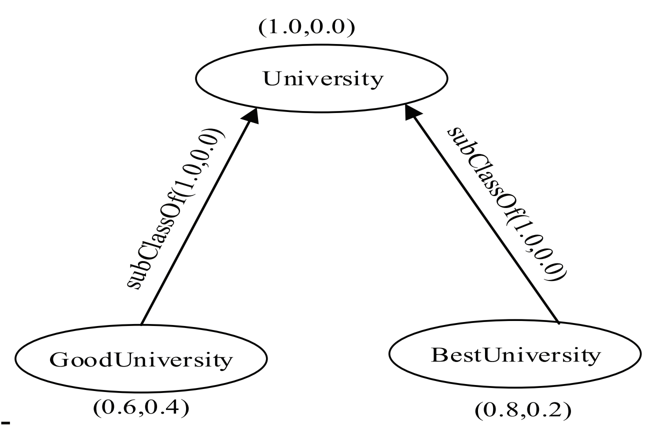

Figure 11. $RDFS_{vg}$ representation.

Definition 3.4: A vague $RDFS_{vg}$ $(RDFS_{vg1} = (R_1, P_1))$ subsumes another vague RDF$(RDFS_{vg2} = (R_2, P_2))$, i.e., $(RDF_{vg}1) \sqsubset (RDFS_{vg}2)$ *iff* $R_1 \sqsubset R_2$, and $P_1 \sqsubset P_2$.

3.4.2. Semantics model of vague RDFS

The semantic model of $RDFS_{vg}$ follows the same guideline as to the semantics of RDF_{vg}. That is, the semantics presented in Hayes and Patel-Schneider (2014) is adopted but modified in such a way that; the set of all resources *IR* comprises of the union of the set of crisp resources (representing crisp classes) denoted as IR_c and the set of vague resources (representing vague classes) denoted as IR_v. That is $IR = IR_c \cup IR_v$. Similarly, the set of crisp properties is represented as IP_c while the set of vague properties is represented as IP_v; therefore, the set of all properties *IP* represents both crisp and vague properties that relate the classes, that is, $IP = IP_c \cup IP_v$. While other conditions are left unchanged.

3.4. Vague graph-based OWL (OWL$_{vg}$)

This section described a vague graph-based OWL (OWL$_{vg}$) by extending the RDFS$_{vg}$ to include some extra constructs and operations for example: the Boolean Combination of classes (such as union, intersection, join, and complement), as well as the capability to express equivalent and disjoint classes, role (property) restriction (such as *oneOf*, *hasValue*, *someValuesFrom*, and *allValuesFrom*)

OWL is based on RDF and RDFS, but it uses different formats such as the basic XML and Turtle (Terse RDF Triple Language) syntaxes to represent the RDF triple models. Since, RDF is a data model for describing resources on the web in form of graph where the nodes represent resources and arcs represent relationships between these resources. Therefore, a graph is a set of statements represented in form of triples comprising of a subject, a predicate, and an object. Having defined our RDF$_{vg}$ and RDFS$_{vg}$, we now describe vague ontologies using the Web ontology language OWL by adopting the method presented in Bourahla (2018) which introduced information on vague statements as annotation assertions within OWL ontologies. Annotation assertions can be defined to conformed with the O_{vg} syntax defined in definition 3.1 of section 3.1.

Some of the properties of OWL ontology that can be used in OWL$_{vg}$ are described in this section as the constructs and operations of OWL$_{vg}$ are defined as follows:

i) **OWL$_{vg}$:** Thing: If *C* is an object in rdfs$_{vg}$, then *C* denotes OWL$_{vg}$: Thing *iff it* contains all objects found in the rdfs$_{vg}$ Ontology.

ii) **OWL$_{vg}$ Nothing:** If C is an object in rdfs$_{vg}$, then *C* denotes OWL$_{vg}$: Nothing *iff* its intersection with any other object of rdfs$_{vg}$ gives that same object.

iii) **Disjoint Classes:** If *C1* and *C2* are objects in rdfs$_{vg}$, then *C1* and *C2* are disjoint if interpretation of *C1* is not equal to the interpretation of *C2*, that is, $I\left(C1,\left(\mathrm{t}_{C1},f_{C1}\right)\right)\cap I\left(C2,\left(\mathrm{t}_{C2},f_{C2}\right)\right)=\varnothing$.

iv) **Equivalent Classes:** If *C1* and *C2* are objects in rdfs$_{vg}$, then *C1* and *C2* are equivalent; if interpretation of *C1* is equal to the interpretation of *C2*, that is, $\left(I\left(C1,\left(\mathrm{t}_{C1},f_{C1}\right)\right)=I\left(C2,\left(\mathrm{t}_{C2},f_{C2}\right)\right)\right)$ or *iff*, there exists a mapping from *C1* to *C2* as well as from *C2* to *C1.*

v) **Individual:** If *C* is a concept in an ontology, then *c* is an instance of a class *C* in rdfs$_{vg}$ and $F(c)=\left(C,\left(\mathrm{t}_{C1},f_{C1}\right)\right)$, where *F* denotes a typing function.

vi) **Intersection of Classes:** Let C1, C2, and C3 represent the sets of objects of RDFS$_{vg}$ (Ob(rdfs$_{vg}$)) such that $c\in C1,C2$ and let P1 and P2 represent the sets of properties associated with Ob(rdfs$_{vg}$) such that $pi\in P1,P2$, then $\mathrm{C3}=\mathrm{C1}\cap\mathrm{C2}=\sigma_{c1}\cap\sigma_{c2},\mu_{p1}\cap\mu_{p2}$ is defined as:

$$t_{\sigma_{c1}\cap\sigma_{c2}}(c)=\left\{\max\left\{t_{\sigma_{c1}}(c),t_{\sigma_{c2}}(c)\right\},\ \textit{if } c\in C1\cap C2\right.$$

$$f_{\sigma_{c1}\cap\sigma_{c2}}(c)=\left\{\max\left\{f_{\sigma_{c1}}(c),f_{\sigma_{c2}}(c)\right\},\quad \textit{if } c\in C1\cap C2\right.$$

$$t_{\mu_{p1}\cap\mu_{p2}}(pi)=\left\{\min\left\{t_{\mu_{pi}}(pi),t_{\mu_{pk}}(pi)\right\},\ \textit{if } pi\in P1\cap P2\right.$$

$$f_{\mu_{p1}\cap\mu_{p2}}(pi)=\left\{\min\left\{f_{\mu_{p1}}(pi),f_{\mu_{p2}}(pi)\right\},\ \textit{if } pi\in P1\cap P2\right.$$

vii) **Union of Classes:** Let C1, C2, and C3 represent the sets of objects of RDFS$_{vg}$ (Ob(rdfs$_{vg}$)) such that $c\in C1,C2$ and let P1 and P2 represent the sets of properties associated with Ob(rdfs$_{vg}$) such that $pi\in P1,P2$, then $\mathrm{C3}=\mathrm{C1}\cup\mathrm{C2}=\sigma_{c1}\cup\sigma_{c2}$, $\mu_{p1}\cup\mu_{p2}$ is defined as:

$$t_{\sigma_{c1}\cup\sigma_{c2}}(c)=\begin{cases} t_{\sigma_{c1}}(c), & \textit{if } c\in C1-C2\\ t_{\sigma_{c2}}(c), & \textit{if } c\in C2-C1\\ \max\left\{t_{\sigma_{c1}}(c),t_{\sigma_{c2}}(c)\right\} & \textit{if } c\in C1\cap C2\end{cases}$$

$$t_{\mu_{p1}\cup\mu_{p2}}(pi)=\begin{cases} t_{\mu_{p1}}(pi), & \textit{if } pi\in P1-P2\\ t_{\mu_{p2}}(pi), & \textit{if } pi\in P2-P1\\ \min\left\{t_{\mu_{p1}}(pi),t_{\mu_{p2}}(pi)\right\} & \textit{if } pi\in P1\cap P2\end{cases}$$

$$f_{\sigma_{c1}\cup\sigma_{c2}}(c)=\begin{cases} f_{\sigma_{c1}}(c), & \textit{if } c\in C1-C2\\ f_{\sigma_{c2}}(c), & \textit{if } c\in C2-C1\\ \max\left\{f_{\sigma_{c1}}(c),f_{\sigma_{c2}}(c)\right\} & \textit{if } c\in C1\cap C2\end{cases}$$

$$f_{\mu_{p1}\cup\mu_{p2}}(pi)=\begin{cases} f_{\mu_{p1}}(pi), & \textit{if } pi\in P1-P2\\ f_{\mu_{p2}}(pi), & \textit{if } pi\in P2-P1\\ \min\left\{f_{\mu_{p1}}(pi),f_{\mu_{p2}}(pi)\right\} & \textit{if } pi\in P1\cap P2\end{cases}$$

viii) **Complement of Classes:** Let C1 and C2 represent the sets of objects of RDFS$_{vg}$ (Ob(rdfs$_{vg}$)), then C1 is a complement of C$_2$ *iff* $I\left(C1,\left(\mathrm{t}_{C1},f_{C1}\right)\right)\cap I\left(C2,\left(\mathrm{t}_{C2},f_{C2}\right)\right)=\varnothing$.

ix) **SubSumption of Classes:** A vague RDF $\left(RDF_{vg}2=\left(R_2,P_2\right)\right)$ subsumes another vague RDF $\left(RDF_{vg}1=\left(R_1,P_1\right)\right)$ represented as $\left(RDF_{vg}1\right)\sqsubseteq\left(RDF_{vg}2\right)$, if $\left(RDF_{vg}1\right)$ is contained in $\left(RDF_{vg}2\right)$, i.e., $R_1\sqsubseteq R_2$ and $P_1\sqsubseteq P_2$.

Example 3.5: The RDF$_{vg}$ in example 3.2 is a sub RDF$_{vg}$ of the RDF$_{vg}$ in example 3.3, since the set of resources and properties (crisp/vague) in example 3.2 are contained in the set of resources and properties (crisp/vague) of example 3.3.

4. Conclusion

In ontology engineering, it is very important to represent uncertainty in order to correctly model most of the real-life domains, especially those that are characterized by uncertainty and imprecision. We have extended the classical SW languages (RDF, RDFS, and OWL) by modeling them as vague graph structures to take into account uncertainty in ontologies. This paper presents RDF_{vg}, $RDFS_{vg,}$ and OWL_{vg}. Since RDF_{vg} and $RDFS_{vg}$ are graphical-based models, their formal definitions and examples were provided. While the OWL_{vg}, being an extension of OWL (which is based on the Description Logic), was described. The uncertain aspect of ontology can be defined as annotations. The idea is based on declaring annotation assertions for vague information using specified annotation property, which will be declared in the OWL_{vg} ontology. These annotation assertions will play the same role to describe vague axioms with the OWL_{vg}. These languages will be applicable in areas, such as medicine, biology, and economics, which are characterized by uncertainty.

There is the need to extend this work in the future to define ontology operations as well as algorithms to achieve the defined operations. Similarly, the inference mechanism under this approach needs to be provided as a part of an extension to this work.

5. Related work

A vocabulary for representing probabilistic relationships in an RDF graph was proposed in Fukushige (2004, 2005) which is to be mapped to a Bayesian network (BN) to perform inference. In the proposed framework, three kinds of probabilistic information can be encoded: probabilistic relations (prior), probabilistic observation (data), and the probabilistic belief (posterior). Any of them can be represented using probabilistic statements that are either conditional or unconditional. The main focus of the work is on getting extended vocabulary that can describe probabilistic relations in a way that is both semantic web compatible and easy to map to a BN, rather than just representing BNs by inventing a new format. A generalized probabilistic RDF was proposed in Udrea et al. (2006). It allows the representation of terminological probabilistic knowledge about classes, as well as *assertional* probabilistic knowledge about the properties of individuals. They proposed a technique based on the idea of logical entailment in probabilistic logic together with local probabilistic semantics for *assertional* probabilistic inference in acyclic probabilistic RDF theory. An algebraic approach to formulizing ontological operations under uncertainty was proposed in Donfack Kana and Akinkunmi (2015), in which algebraic theories of uncertainty were used to formalize operations within and between ontologies. A soft set theory was used to define a set-like view of ontology and to approximate its concepts of instances. In order to handle uncertainty, rough set theory was used to measure the degree at which an individual is an instance of a given concept. A model called ontologies of uncertainty (OU) was proposed to handle both crisp and vague aspect of the domain of discourse by classifying attributes of ontologies into sets of vague attributes (Av) and crisp attributes (Ac). While relations of ontologies were also classified into sets of vague relations (Rv) and crisp relations (Rc).

Bourahla (2015) presented a vagueness theory to deal with the problem of ontologies that contain vague concepts, whereas the concepts are treated as having a meaning in such a way that the vague concept evolves during the ontology evolution, from more vague meaning to less vague meaning until it reaches a situation where it becomes a non-vague concept as opposed to having a fixed meaning that is shared by all users of the ontology. A language for vagueness description in Web ontologies called VDL was presented in Bourahla (2018) to represent vague axioms that can be found in ontologies. The vagueness descriptions are used as inputs to a reasoning procedure (VR) developed and implemented using the Prolog language (Logic Programming) to enable reasoning over ontologies that have vague axioms.

References

Bourahla M. Reasoning over vague concepts. In International Conference on Artificial Intelligence and Soft Computing (ICAISC), Zakopane, Poland, (LNCS, 9120), 2015, pp 591–602; doi: 10.1007/978-3-319-19369-4_52

Bourahla M. Description and reasoning for vague ontologies using logic programming. IET Software Special Issue: Software Architecture for the Web of Things (SAWoT), v()p–, 2018. ISSN 1751-8806, doi:10.1049/iet-sen.2017.0226

Donfack Kana AF, Akinkunmi BO. Formalising ontological operations under uncertainty using algebraic techniques. An unpublished PhD. Thesis submitted to the Department of Computer Science, Faculty of Science. University of Ibadan, Nigeria, 2015.

Fukushige Y. Representing probabilistic knowledge in the semantic web, Cambridge, MA. In Position paper for the W3C Workshop on Se-mantic Web for Life Sciences, 2004.

Fukushige Y. Representing probabilistic relations in RDF. In Proceedings of Work-shop on Uncertainty Reasoning for the Semantic Web (URSW) at the 4th International Semantic Web Conference (ISWC), Galway, Ireland, 2005.

Kenneth J, Kathryn BL, Paulo CGC, Miecyslaw MK, Trevor M, Thomas L. Uncertainty reasoning for the World Wide Web (W3C) Incubator Group Report, 2008. Available via https://www.w3.org/2005/Incubator/urw3/XGR-urw3/#model

Ramakrishna N. Vague graphs. Int J Comput Cogn 2009; 7:51–8.

Udrea O, Subrahmanian V, Majkic Z. Probabilistic RDF. In Proceedings of the IEEE International Conference on Information Reuse and Integration; Heuristic Systems Engineering. IEEE Systems, Man, and Cybernetics Society. Waikoloa, Hawaii, pp 172–7, 2006.

Zhongli D, Yun P, Rong P. BayesOWL: uncertainty modeling in semantic web ontologies. In Soft Computing in Ontologies and Semantic Web. Springer Berlin / Heidelberg, vol. 204, pp 3–29, 2006; doi:10.1007/978-3-540-33473-61

Hayes, P. J. and Patel-Schneider, P. F. (2014). Rdf 1.1 semantics. W3C Recommendation, Online at http://www.w3. org/TR/rdf11-mt.