Possibilistic Extension of Domain Information System (DIS) Framework

Deemah Alomair[1,2][a] and Ridha Khedri[1][b]
[1] *Department of Computing and Software, McMaster University, Hamilton, Canada*
[2] *Department of Computer Information Systems, Imam Abdulrahman Bin Faisal University, K.S.A.*

Keywords: Uncertainty, Incomplete Information, Knowledge Representation and Reasoning, Ontology Modelling, Ontologies, Possibilistic Logic, Ontology Reasoning.

Abstract: Uncertainty poses a significant challenge in ontology-based systems, manifesting in forms such as incomplete information, imprecision, vagueness, ambiguity, or inconsistency. This paper addresses this challenge by introducing a quantitative possibilistic approach to manage and model incomplete information systematically. Ontologies are modelled using the Domain Information System (DIS) framework, which is designed to handle Cartesian data structured as sets of tuples or lists, enabling the construction of ontologies grounded in the dataset under consideration. Possibility theory is employed to extend the DIS framework, enhancing its ability to represent and reason with incomplete information. The proposed extension captures uncertainty associated with instances, attributes, relationships, and concepts. Furthermore, we propose a reasoning mechanism within DIS that leverages necessity-based possibilistic logic to draw inferences under uncertainty. The proposed approach is characterized by its simplicity. It improves the expressiveness of DIS-based systems, introducing a foundation for flexible and robust decision-making in the presence of incomplete information.

1 INTRODUCTION

One of the primary challenges in knowledge-based systems, particularly those that rely on ontologies for domain reasoning, is managing uncertainty stemming from incomplete information. In dataset-driven ontologies, data is contextualized to define concepts, relationships, and instances. However, real-world applications frequently suffer from missing or partial information, leading to epistemic uncertainty (Sentz and Ferson, 2002). This type of uncertainty affects instance classification, attribute reliability, relationship strength, and concept validity. When unaddressed, such uncertainty can render ontologies either overly rigid, failing to accommodate partial knowledge, or misleading, by permitting unjustified inferences. Effectively managing uncertainty is therefore essential to ensure the expressiveness, reliability, and adaptability of ontology-based systems, especially in the context of decision support or automated reasoning systems. To illustrate, consider a customer service ontology; the concept `PositiveFeedback` may depend on attributes like `Satisfaction`, `Quality`, and `ResponseTime`. If one of these values is missing or partially available, classical inference systems may fail to classify an instance as `PositiveFeedback` or do so incorrectly. This highlights the need for a framework that can represent and reason under partial knowledge.

This paper introduces a quantitative possibilistic extension to the Domain Information System (DIS) framework (Marinache et al., 2021) to represent and reason under partial knowledge. DIS is a bottom-up, data-centric formalism that constructs ontologies from datasets, structurally separating the domain ontology from the data view and linking them via a mapping operator. Unlike Description Logic (DL)-based ontologies, which separate the A-Box and T-Box logically, DIS achieves this separation structurally and grounds the ontology in data, reducing data-ontology mismatches. DIS is useful for aligning ontologies with real-world datasets, which makes it particularly effective for domains where ontologies must be generated or adapted from existing data sources, improving modularity, transparency, and maintainability in ontology design. In contrast to traditional ontology languages like Web Ontology Language (OWL), which struggle to directly represent mereological relationships in Cartesian datasets (i.e., the structured data itself) without complex extensions, DIS lever-

[a] https://orcid.org/0000-0001-9397-9999
[b] https://orcid.org/0000-0003-2499-1040

Alomair, D. and Khedri, R.
Possibilistic Extension of Domain Information System (DIS) Framework.
DOI: 10.5220/0013706800004000

In *Proceedings of the 17th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2025) - Volume 2: KEOD and KMIS*, pages 108-119
ISBN: 978-989-758-769-6; ISSN: 2184-3228

ages cylindric algebra and Boolean algebra to model both data structures and conceptual part-whole relations. This enables more natural and robust handling of mereological reasoning within structured data. However, the original DIS model does not capture domain uncertainty and information uncertainty. The proposed approach overcomes this by associating each ontological component with quantified certainty.

Unlike vagueness or imprecision, the focus here is on uncertainty due to incompleteness, typically addressed via probability theory (e.g., (Laha and Rohatgi, 2020)), possibility theory (e.g., (Dubois and Prade, 2015)), or Dempster–Shafer theory (e.g., (Sentz and Ferson, 2002)), as discussed in (Alomair et al., 2025). In this study, possibility theory is adopted and rationale behind this selection is explained in the section 5.

The proposed approach models uncertainty across all key ontological elements: attributes, concepts, relationships, and instances. The key contributions of this paper are as follows:

1. Modelling Uncertainty of Attributes: Introduces a necessity-based mapping from the dataset's attributes to ontology concepts.

2. Modelling Uncertainty of Instances: Proposes an instance distribution relation ($SV^{\mathcal{D}}$), allowing a datum (instance) to be assigned to multiple sorts (attributes) with varying degrees of certainty.

3. Modelling Uncertainty of Relationships: Introduces necessity-based relationship, which allows relationships to hold with varying levels of certainty.

4. Modelling Uncertainty of Concepts: Refines the construction of datascape concepts (which depend on available data values) by incorporating uncertainty modelling into their data-specializing predicate.

5. Possibilistic Reasoning for Uncertainty-Aware Inference: Develops a reasoning mechanism within the DIS framework, leveraging necessity-based possibilistic logic to support inference under incomplete information.

The paper is structured as follows: Section 2 introduces foundational theories. Section 3 presents the integration of possibilistic components into the DIS framework, followed by uncertainty-aware reasoning in Section 4. Section 5 reviews related work and offers a discussion. Section 6 concludes the paper and outlines future directions.

2 PRELIMINARIES

This section reviews uncertainty in ontology, introduces possibility theory and possibilistic logic, and presents the theoretical background of the DIS framework.

2.1 Uncertainty and Ontology

Information imperfection includes incompleteness, imprecision, vagueness, ambiguity, and inconsistency (Ma et al., 2013; Bosc and Prade, 1997). The paper adopts a broad interpretation, considering uncertainty as arising from any of these deficiencies, as adopted in (Anand and Kumar, 2022; Ceravolo et al., 2008). *Incompleteness* arises when information is partial. This creates uncertainty about which interpretation of a statement to rely on, often addressed by calculating an estimation degree for possible worlds (Straccia, 2013). *Imprecision* refers to the lack of exactness, occurring when data is expressed in approximate or qualitative terms instead of precise values (Ma et al., 2013). *Vagueness* emerges when terms or concepts lack clear boundaries (Straccia and Bobillo, 2017). *Ambiguity* arises from multiple interpretations (Ma et al., 2013), and *inconsistency* involves contradictions, such as conflicting statements (Bosc and Prade, 1997).

An extensive review of uncertainty modelling in domain ontologies is presented in (Alomair et al., 2025). The survey examines over 550 studies published between 2010 and 2024 on this topic. A guiding taxonomy is proposed, classifying ontological uncertainty into *concept uncertainty* and *information uncertainty*. This classification supports the systematic identification of uncertainty types across ontological frameworks and the selection of appropriate formalisms to manage them. *Concept uncertainty* involves uncertainty of relationships, uncertainty of attributes defining a concept, and uncertainty due to semantic ambiguity, where context influences the interpretation of a concept. *Information uncertainty* concerns associating instances with concepts or relations. The identified uncertainties are attributed to incomplete, imprecise, vague, or inconsistent information. Then, various formalisms are presented to manage these uncertainties. This taxonomy offers a structured approach to understanding and addressing uncertainty in ontology-driven systems. A visual representation of the taxonomy is shown in Figure 1.

2.2 Possibility Theory and Possibilistic Logic

Possibility theory models incomplete and inconsistent knowledge using qualitative (ordinal) or quantitative (numerical) approaches (Dubois and Prade, 2015).

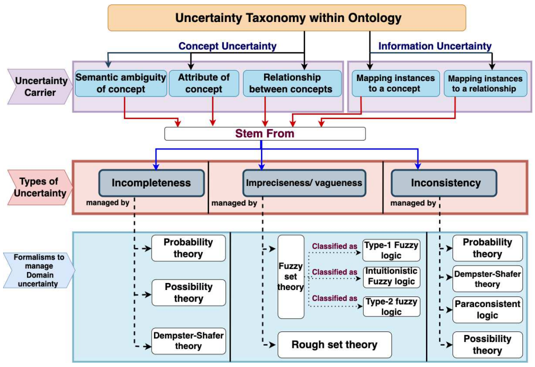

Figure 1: Ontological Uncertainty Taxonomy.

The qualitative approach ranks events without numerical degree (e.g., ”highly possible”, ”possible”, or ”less possible”), while the quantitative approach assigns a numerical degree to represent degrees of possibility. Possibility distribution represents an agent's knowledge about the world by assigning plausibility degrees to states in a set S, which may be finite or infinite. Formally, it is a function $\pi : S \to L$, where L is a totally ordered scale (often $[0,1]$). The value $\pi(x)$ expresses how plausible the state $x \in S$. A value of $\pi(x) = 0$ means state x is impossible, while $\pi(x) = 1$ means it is fully plausible. If S is exhaustive, at least one state must have plausibility 1. The possibilistic framework captures both complete and incomplete knowledge. Complete knowledge is represented by assigning possibility 1 to a single state and 0 to all others. Complete ignorance is modeled by assigning possibility 1 to all states, indicating that any state could be true. The possibility distribution forms the basis for defining possibility and necessity measures over any subset $X \subseteq S$:

$$\Pi(X) = \sup_{x \in X} \pi(x) \text{ and } N(X) = \inf_{x \notin X}(1 - \pi(x)), \quad (1)$$

where $\Pi(X)$ indicates feasibility, and $N(X)$ expresses certainty (Alola et al., 2013). The measures are dual via: $N(X) = 1 - \Pi(X')$, where X' is the complement of X. Possibility measures follow the *maxitivity axiom*: $\Pi(A \cup B) = \mathtt{max}(\Pi(A), \Pi(B))$, while necessity measure satisfies the dual *minitivity axiom*: $N(A \cap B) = \mathtt{min}(N(A), N(B))$. The necessity degree for the union of two sets satisfies the following property, expressed as (Dubois and Prade, 2014):

$$N(A \cup B) \geq \mathtt{max}(N(A), N(B)) \quad (2)$$

Unlike probability theory, which quantifies likelihood, possibility theory evaluates feasibility. In (Zadeh, 1999), a distinction between possibility and probability theories has been made through an example of *”Hans is eating eggs for breakfast”*. In his example, the possibility distribution of ($\pi_X(3) = 1$) suggests it is entirely possible for Hans to eat three eggs, but the probability ($P_X(3) = 0.1$) indicates this outcome is statistically rare. This demonstrates that high possibility does not imply high probability, though an impossible event ($\pi_X(u) = 0$) has zero probability ($P_X(u) = 0$).

Possibility theory underpins possibilistic logic, which we limit here to necessity-based possibilistic logic (Dubois et al., 1994; Dubois and Prade, 2014; Nieves et al., 2007). In this logic, a formula is a pair (θ, α), where θ is a classical first-order logic formula, and $\alpha \in [0,1]$ is a certainty or priority degree. This pair indicates that θ is certain at least to level α, (i.e., $N(\theta) \geq \alpha$). The interval $[0,1]$ can be replaced by any linearly ordered scale. Standard limit conditions hold: $\Pi(\bot) = N(\bot) = 0$, $\Pi(\top) = N(\top) = 1$, where $\bot$ and $\top$ denote contradiction and tautology, respectively. In the formal system of this logic, the following properties hold: $N(\theta \wedge \gamma) = \mathtt{min}(\{N(\theta), N(\gamma)\})$ and $N(\theta \vee \gamma) \geq \mathtt{max}(\{N(\theta), N(\gamma)\})$, where θ and γ are formulae. One of its main rules is the weakest link resolution rule:

$$(\neg\theta \vee \gamma, \alpha), (\theta \vee \delta, \beta) \vdash (\gamma \vee \delta, \mathtt{min}(\alpha, \beta)), \quad (3)$$

Here, the conclusion's certainty is the smallest among the premises, reflecting that an inference chain is limited by its weakest premise.

The weighted minimum and maximum operations, introduced in (Grabisch, 1998; Dubois and Prade, 1986) within the framework of possibility theory, generalize the standard `min` and `max` functions to account for elements from different contexts, each associated with a distinct weight of importance. These operations refine aggregation by modulating the influence of each element based on its assigned weight.

Let $X = \{x_1, \cdots, x_n\}$ be a set of criteria. Let a_i and w_i be, respectively, the score and the weight of importance attributed to criterion x_i such that $\Sigma_{i=1}^{n} w_i = 1$. Then we have:

$$\mathtt{Weighted\_Min}(a_1, \cdots, a_n) = \mathtt{min}(i \mid 1 \leq i \leq n : \mathtt{max}((1 - w_i), a_i)).$$

This formulation ensures that elements with lower weights contribute less to the overall minimum computation. Similarly, the operation `Weighted_Max` is given by:

$$\mathtt{Weighted\_Max}(a_1, \cdots, a_n) = \mathtt{max}(i \mid 1 \leq i \leq n : \mathtt{min}(w_i, a_i)).$$

2.3 Domain Information System

DIS is an ontology framework consisting of three primary components (Marinache et al., 2021; Marinache, 2025): Domain Ontology View (DOnt) $\mathcal{O}$, Domain Data View (DDV) $\mathcal{A}$, and mapping function τ linking $\mathcal{A}$ to $\mathcal{O}$, forming the structure $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$.

The DOnt, $\mathcal{O} = (\mathcal{C}, \mathcal{L}, \mathcal{G})$, is composed of three elements. The concept structure $\mathcal{C} = (C, \oplus, e_c)$, which is a commutative idempotent monoid where the carrier set C includes an empty concept (e_c), a set of atomic concepts ($\mathcal{T}$) derived directly from dataset attributes, and composite concepts formed using the $\oplus$ operator. The Boolean lattice $\mathcal{L} = (L, \sqsubseteq_c)$ organizes concepts hierarchically based on a natural order $\sqsubseteq_c$, defined as $c_1 \sqsubseteq_c c_2 \iff c_1 \oplus c_2 = c_2$. Lastly, the set of rooted graphs $\mathcal{G}$ provides additional expressiveness by capturing concepts and relations beyond those defined by the lattice structure. Each rooted graph $G_{t_i} = (C_i, R_i, t_i)$ consists of a set of vertices $C_i \subseteq C$, a set of edges R_i, and a root vertex $t_i \in L$.

The DDV, $\mathcal{A} = (A, +, \star, -, 0_A, 1_A, \{c_k\}_{k \in \mathcal{U}})$, is formalized as a diagonal-free cylindric algebra, where $\mathcal{U}$ is a finite set of sorts (the universe). The main notion of this view is sort, which corresponds to an attribute in the dataset. The ordered pair of a sort and its value is known as `Sorted Value` (`SV`). A set of `SV` with a maximum of one `SV` for each sort forms `Sorted Datum` (`S_Datum`). The carrier set A consists of `Sorted Data` (`S_Data`), structured as a set of `S_Datum`. The cylindrification operators c_k are indexed by the sorts used in the data, corresponding to the elements of L, the carrier set of the Boolean lattice $\mathcal{L}$. For a deeper understanding of cylindric algebra, readers are referred to (Imieliński and Lipski, 1984).

The final component of DIS is the mapping function $\tau : A \to L$, which links the elements of $\mathcal{A}$ in DDV to their corresponding concepts in the Boolean lattice $\mathcal{L}$ within DOnt. To define τ, several helper operators introduced, one of which is the helper mapping operator $\eta : \mathcal{U} \to L$. This ensures a one-to-one correspondence between the sorts in DDV and the atomic concepts in the Boolean lattice of DOnt. Ensuring a seamless mapping from data attributes to ontology concepts: $\eta(S_{attr}) = attr$, where S_{attr} and $attr$ are a sort and an atomic concept, respectively.

In DIS, concepts are categorized based on their dependence on objective reality or data elements, leading to the distinction between *objective concepts* and *datascape concepts*, denoted by C_d. Objective concepts exist independently of any dataset. For instance, consider the objective statement $\exists(x \mid x \in \text{Animal} : \text{Pet}(x))$. The concept *Pet* remains valid regardless of whether supporting data is available. In contrast, *datascape concepts* rely on data for their definition and existence. For instance, consider the modified example $\exists(x \mid x \in \text{Animal} : \text{Active\_Pet}(x))$. The concept *Active_Pet*, defined as a pet that exercises for at least one hour daily, depends on a specific data source such as daily activity logs. If such data is unavailable or does not meet the required conditions, the concept cannot be realized. Formally, a datascape concept in a DIS is defined as follows:

Definition 1 (From (Alomair and Khedri, 2025), Datascape Concept)**.** *Let $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$ be a given DIS. For a carrier set A in $\mathcal{A}$ and a lattice $\mathcal{L}$ in $\mathcal{O}$, a datascape concept C_d is defined as follows: $C_d \stackrel{def}{=} \{a \mid \tau(a) \in L \wedge \Phi(a)\}$, where $a \in A$ and Φ is a data-specializing predicate expressed in Disjunctive Normal Form (DNF). This predicate Φ is given by: $\Phi(a) = \vee(i \mid 1 \le i \le N : \Psi_i(a))$, with N is a natural number, and each conjunctive clause $\Psi_i(a)$ is defined as: $\Psi_i(a) = \wedge(j \mid 1 \le j \le M : \Omega_{(i,j)}(a))$, where M is a natural number and $\Omega_{(i,j)}(a) = (f_{(i,j)}(a \cdot \mathtt{sort\_name}_{(i,j)}), c_{(i,j)}) \in R_{(i,j)}$, where $f_{(i,j)} \in \mathcal{F}$, and $\mathcal{F} = \{\oplus, e_c, \top_{\mathcal{L}}, +, \star, -, 0, 1, \tau, cyl\}$ is the set of function symbols, $c_{(i,j)}$ is a ground term in the DIS language, and $R_{(i,j)}$ is a relator.* $\square$

Based on the above definition, we define the operation $\oplus$ as an operation on concepts.

Definition 2. *Let $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$ be a given DIS. Let $C_{d1} = \{a \mid \tau(a) \in L \wedge \Phi_1(a)\}$, and $C_{d2} = \{a \mid \tau(a) \in L \wedge \Phi_2(a)\}$ be two datascape concepts defined on $\mathcal{D}$. We have $C_{d1} \oplus C_{d2} =$*

$$C_{d1} \cup C_{d2} = \{a \mid \tau(a) \in L \wedge (\Phi_1(a) \vee \Phi_2(a))\}.$$

The structure of the $C_{d1} \oplus C_{d2}$ is that of a datascape as $(\Phi_1(a) \vee \Phi_2(a))$ is in DNF and the other conditions stipulated by Definition 1 are satisfied. Moreover, the empty concept e can be perceived as a datascape concept defined as $e = \{a \mid \tau(a) = e_c \wedge \mathsf{false}\} = \emptyset$. Hence, if we take, for a given DIS, C_d is the set of datascape concepts, then $(C_d, \oplus, e_c)$ is a commutative monoid due to the properties of set union.

Illustrative Example of DIS Construction. We consider a `CustomerService` dataset with the attributes: `Satisfaction`, `Quality`, and `ResponseTime`. The corresponding DIS structure is built as follows:

1. Lattice construction: Each dataset attribute is mapped to an atomic concept: $\tau = \{(\mathtt{Quality}, \mathtt{Status}), (\mathtt{ResponseTime}, \mathtt{Duration}), (\mathtt{Satisfaction}, \mathtt{Comfort})\}$. Then the rest of the Boolean lattice is generated, where each node represents a possible composition of atomic concepts (e.g., $\mathtt{status\ Tenure} = \mathtt{Status} \oplus \mathtt{Duration}$).

2. Objective rooted graph concept: Rooted graphs enrich the ontology beyond lattice nodes. One such objective concept is `Feedback`, rooted at `CustomerService`, and defined abstractly as follows: `Feedback` $\overset{\text{def}}{=} \{a \mid \tau(a) \in$ `CustomerService`$\}$.
3. Datascape rooted graph concept: A rooted graph concept might be a datascape concept, in which its definition depends on data. For example, the concept `PositiveFeedback` can be defined as: `PositiveFeedback`$= \{a \mid \tau(a) \in$ `CustomerService` $\wedge\ a.$`Satisfaction` $\geq 0.6\}$. The predicate here indicates that an instance a of the `Satisfaction` attributes should have a value greater than or equal to 0.6.
4. Construction of the domain data view: An example of `SV` is (`Quality`, `Good`). An example of `S_Datum` is $dt\_1 = \{$(`Quality`, `Good`), (`ResponseTime`, `Fast`), (`Satisfaction`, `Yes`)$\}$. An example of `S_Data` is $a = \{dt\_1, dt\_n\}$.
5. Building the whole DIS system: The DIS is then formed by $(\mathcal{O}, \mathcal{A}, \tau)$. A full illustration of the DIS structure is shown in Figure 2.

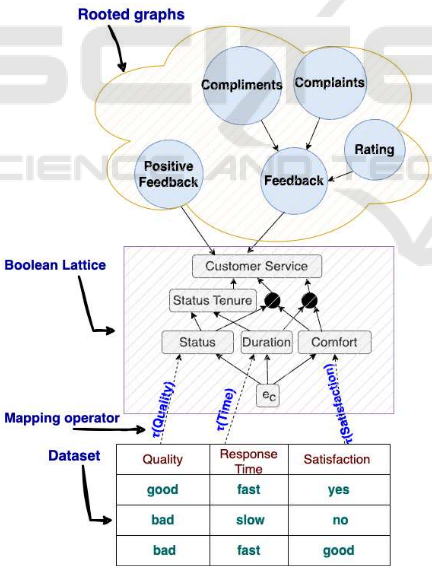

Figure 2: Customer Service DIS Framework.

3 UNCERTAINTY MODELLING IN DIS FRAMEWORK

In this section, we extend the DIS framework to handle uncertainty by addressing two key questions: What type of uncertainty can be modelled, and where in the DIS framework it can be introduced. As noted in section 1, we focus on incomplete information and adopt possibility theory as the formalism.

To illustrate where uncertainty can arise, Figure 3 shows an example using a customer service dataset. Database attributes may assign values, introducing uncertainty of instances. These attributes are mapped via the operator τ (shown by arrows) to atomic concepts introducing uncertainty of attributes. The lattice is further expanded with multiple rooted graphs, such as `PositiveFeedback` and `Feedback`, introducing uncertainty of concepts. The `Feedback` graph includes specialized concepts like `Rating`, with arrows indicating semantic paths among these concepts, capturing the uncertainty of relationships.

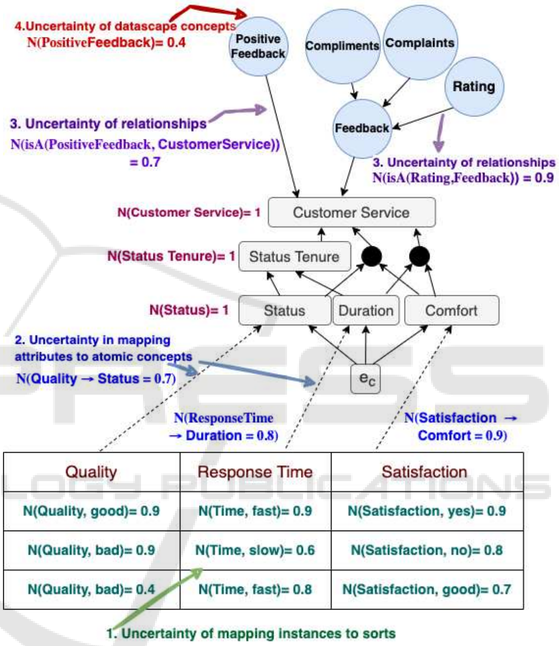

Figure 3: Necessity Degrees Assigned to DIS.

Since the focus is on uncertainty due to incomplete information, it is crucial to distinguish between data and information. In our formalism, a datum is strictly a raw value without any assigned context (e.g., the number 3.7 isolated from metadata, units, or semantics). At this stage, it has no uncertainty; Uncertainty arises only when contextual interpretation is applied (e.g., labelling 3.7 as "sensor voltage reading with ±0.2 error"). We acknowledge that the broader literature often treats data as implicitly contextualized (and thus uncertain), but our formalism explicitly separates raw values from their contextual layers. It is also important to emphasize that the assigned degree is explicitly interpreted as a measure of certainty, not as a degree of truth or graded quality. For this reason, we adopt necessity-based possibilistic logic, where necessity degrees directly correspond to the degree of certainty. This interpretation aligns naturally with our setting, in which the degree reflects the certainty in the existence of concepts, in instance-to-concept and attribute-to-concept associations, and in the presence of relationships. In our approach, we examine four types of uncertainty:

1. *Uncertainty of mapping instances to sorts*: When mapping a value to a sort(attribute), for example, as indicated in Figure 3, the $\texttt{Quality}$ attribute being assigned values like $\texttt{Good}$ or $\texttt{Bad}$, with a necessity degree reflecting the degree of certainty with which the value belongs to a given sort. For instance, assigning $(\texttt{Good}, 0.9)$ to $\texttt{Quality}$ indicates that for this particular instance, it is 0.9 certain that the $\texttt{Quality}$ is $\texttt{Good}$.

2. *Uncertainty in mapping attributes to atomic concepts*: When mapping a sort to a lattice concept, such as associating $\texttt{Quality}$ with the $\texttt{Status}$ concept as $N(\texttt{Quality} \rightarrow \texttt{Status}) = 0.7$.

3. *Uncertainty of relationships*: When defining relationships among rooted graph concepts, like the relationship between $\texttt{Rating}$ and $\texttt{Feedback}$ is associated with $N(\texttt{isA}(\texttt{Rating}, \texttt{Feedback})) = 0.9$.

4. *Uncertainty of datascape concepts*: This uncertainty arises when a concept is defined in terms of data conditions that may themselves be uncertain. For example, consider the datascape concept $\texttt{PositiveFeedback}$, defined as $\texttt{PositiveFeedback} = \{a \mid \tau(a) \in \texttt{CustomerService} \wedge a.\texttt{Satisfaction} \geq 0.6\}$. Here, the condition $(\Phi(a) = a.\texttt{Satisfaction} \geq 0.6)$ is the data-specializing predicate that characterizes the concept. In our framework, the necessity degree of the datascape concept itself, that is, the degree to which the concept $\texttt{PositiveFeedback}$ holds in the presence of incomplete information, is derived directly from the necessity with which its data-specializing predicate is satisfied.

The first three types of uncertainty that are listed above are given by the domain expert, while the last one is calculated.

3.1 Uncertainty of Mapping Instances to Sorts

In the traditional DIS framework, data records (instances) are typically assigned to sorts (attributes) through a certain mapping function. This assignment is $\texttt{SV} : V \rightarrow \mathcal{U}$, where V is a finite set of values assigned to the sort, and $\mathcal{U}$ is a finite set of sorts (the universe). For example, consider a customer service database presented in Figure 3, where the attribute $\texttt{Quality}$ can take values such as $\texttt{Good}$ and $\texttt{Bad}$. The traditional mapping function would assign these values to the $\texttt{Quality}$ sort, as $\texttt{SV}(\texttt{Good}) = \texttt{Quality}$ and $\texttt{SV}(\texttt{Bad}) = \texttt{Quality}$. However, uncertainty brings nondeterminism in this mapping, as a value might be assigned to several sorts with some degree of certainty. Hence, to account for the uncertainty in these assignments, we introduce a new relation called the *instance distribution relation*, denoted $SV^{\mathcal{D}}$, and defined as $SV^{\mathcal{D}} \subseteq V \times \mathcal{U} \times [0,1]$. The relation $SV^{\mathcal{D}}$ relates a data value to sorts and necessity degree that represent the degree of certainty in the assignment. A data value might be assigned to several sorts with varying degrees of certainty. In the example, the instance distribution relation could return values like: $SV^{\mathcal{D}} = \{\, (\texttt{Good}, (\texttt{Quality}, 0.9)), (\texttt{Bad}, (\texttt{Quality}, 0.9)), (\texttt{Bad}, (\texttt{Quality}, 0.4)), (\texttt{Good}, (\texttt{Satisfaction}, 0.7))\}$. Here, the data value $\texttt{Good}$ is assigned to the $\texttt{Quality}$ sort with a certainty of 0.9, while the value $\texttt{Bad}$ is assigned to the same sort with two different certainty degrees: 0.9 and 0.4. These reflect varying contexts, such as different data records, where assignment certainty differs. Although the notation does not explicitly represent context, it is implicitly captured through association with different instances. Additionally, $\texttt{Good}$ is assigned to the $\texttt{Satisfaction}$ sort with a certainty of 0.7. This extension enables the framework to better reflect uncertainty by accommodating varying degrees of certainty in data-to-sort assignments.

3.2 Uncertainty in Mapping Attributes to Atomic Concepts

As previously discussed in subsection 2.3, the DIS framework defines the helper mapping operator $\eta : \mathcal{U} \rightarrow L$, which assigns each sort (attributes) in $\mathcal{U}$ to its corresponding atomic concept in the Boolean lattice. Similar to the *uncertainty of instances*, uncertainty in attribute mapping introduces non-determinism when associating sorts with atomic lattice concepts. To account for this, we define the *mapping distribution relation* $\eta^{\mathcal{D}}$, which captures the uncertainty in this mapping. The relation $\eta^{\mathcal{D}} \subseteq \mathcal{U} \times L \times [0,1]$ is provided by a domain expert, and assigns a necessity degree to each potential mapping.

Consider the customer service database presented in Figure 3, where the mapping operators η are defined as follows:

$$\eta(\texttt{Quality}) = \texttt{Status}, \eta(\texttt{Satisfaction}) = \texttt{Comfort}, \eta(\texttt{ResponseTime}) = \texttt{Duration}.$$

In this mapping, the sorts `Quality`, `Satisfaction`, and `ResponseTime` correspond to the atomic concepts `Status`, `Comfort`, and `Duration`, respectively. To capture the uncertainty in these mappings, the mapping distribution relation $\eta^{\mathcal{D}}$ assigns a necessity degree to each association:

$\eta^{\mathcal{D}}(\texttt{Quality}) = (\texttt{Status},\ 0.7)$,
$\eta^{\mathcal{D}}(\texttt{Satisfaction}) = (\texttt{Comfort},\ 0.9)$,
$\eta^{\mathcal{D}}(\texttt{ResponseTime}) = (\texttt{Duration},\ 0.8)$.

These degrees indicate the degree of certainty in each mapping, allowing the DIS framework to handle the uncertainty in the alignment between data attributes and ontology concepts.

3.3 Uncertainty of Relationships

Within the DIS framework, there are relationships between the concepts of rooted graphs and a parthood relationship between the concepts of the Boolean lattice. The parthood relationship $\sqsubseteq_c$ forms the relationship between objective concepts given in the lattice. The existence of this relationship among lattice concepts is certain, as they are constructed by a Cartesian construction from the atomic concepts. In other terms, a concept k_1 is considered a `partOf` another concept k, if k_1 is a Cartesian projection of k or if its atomic structure is a subset of that of k. However, the relations among the concepts of the rooted graph might be uncertain. Given a rooted graph $G_{t_i} = (C_i, R_i, t_i)$, $C_i \subseteq C$, $R_i \subseteq C_i \times C_i$, $t_i \in L$, its relation is transformed to give each edge a necessity degree. We extend R_i to a necessity-based $R_i^{\mathcal{D}}$. Hence, $R_i^{\mathcal{D}} \subseteq R_i \times [0,1]$, which incorporates necessity degrees to quantify the degree of certainty associated with each relationship.

In the customer service database illustrated in Figure 3, the relation of the rooted graph, denoted by R_i, is the following: $R_i =$

$\{(\texttt{isA(PositiveFeedback,CustomerService)}),$
$(\texttt{isA(Complaints,Feedback)}),$
$(\texttt{isA(Rating,Feedback)}),$
$(\texttt{isA(Feedback,CustomerService)}),$
$(\texttt{isA(Compliments,Feedback)})\}$

Hence, the relations $R_i^{\mathcal{D}}$ is given as follows: $R_i^{\mathcal{D}} = \{$
$(\texttt{isA(PositiveFeedback,CustomerService)}, 0.4),$
$(\texttt{isA(Complaints,Feedback)}, 1),$
$(\texttt{isA(Rating,Feedback)}, 0.9)$
$(\texttt{isA(Feedback,CustomerService)}, 0.5),$
$(\texttt{isA(Compliments,Feedback)}, 0.7)\}$

These necessity degrees quantify the degree of certainty in each relationship, enabling the framework DIS to systematically capture and reason about uncertainty in relational structures.

3.4 Uncertainty of Datascape Concepts

If we examine the elementary predicate $\Omega_{(i,j)}(a)$, which is used in building the data-specializing predicate $\Phi(a)$ of a datascape concept and which is equal to $(f_{(i,j)}(a \cdot \texttt{sort\_name}_{(i,j)}), c_{(i,j)}) \in R_{(i,j)}$, we find that there are two sources of uncertainty. The first comes from mapping a datum a to a sort due to the usage of the term $a \cdot \texttt{sort\_name}_{(i,j)}$, and the second comes from the relator $R_{(i,j)}$ used in $\Omega_{(i,j)}(a)$. Hence, by capturing these two sources of uncertainty, we capture the uncertainty of the datascape concept. For that, we adopt the weighted minimum function, previously defined in subsection 2.2. The weights of instance mapping $w_{\texttt{inst}}$, and the weight of the relationship $w_{\texttt{rel}}$ assign relative importance to the necessity measures $SV^{\mathcal{D}}(a)$ and $R_i^{\mathcal{D}}(a)$, with $w_{\texttt{inst}} + w_{\texttt{rel}} = 1$. Then, we have the following inductive procedure for calculating the necessity degree $N_{\Phi(a)}$ of a datascape concept having Φ as its data-specializing predicate.

Procedure 3.1 (Necessity Degree of a Datascape Predicate)**.** *Let $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$ be a given DIS. Let $C_d = \{a \mid \tau(a) \in L \wedge \Phi(a)\}$ be a datascape concept that is defined within $\mathcal{D}$, and has Φ as its specializing predicate. For a given element $a \in A$, let $\delta = (w_{inst} = w_{Wrel}) \vee \Big((SV^{\mathcal{D}}(a) \leq (1 - w_{inst})) \wedge (R^{\mathcal{D}}_{(i,j)}(a) \leq (1 - w_{Wrel}))\Big)$. The necessity degree $N(\Phi(a))$ is computed inductively as follows:*

- *Base cases:*
  1. $N(\mathit{true}) = 1$;
  2. $N(\mathit{false}) = 0$.
  3. $N(\Omega_{(i,j)}(a)) =$

$$\begin{cases} \mathtt{min}\Big(SV^{\mathcal{D}}(a), R^{\mathcal{D}}_{(i,j)}(a)\Big), & \text{if } \delta = \mathit{true}, \\ \mathtt{min}\Big(\mathtt{max}\Big((1 - w_{inst}), SV^{\mathcal{D}}(a)\Big), \\ \quad \mathtt{max}\Big((1 - w_{rel}), R^{\mathcal{D}}_{(i,j)}(a)\Big)\Big), & \text{otherwise}. \end{cases}$$

- *Inductive cases:*
  1. *Conjunction of atomic predicates:*

$$N(\Psi_i(a)) = \mathtt{min}\Big(j \mid 1 \leq j \leq M : N(\Omega_{(i,j)}(a))\Big)$$

  2. *Disjunction of conjunctive clauses:*

$$N(\Phi(a)) = \mathtt{max}\Big(i \mid 1 \leq i \leq N : N(\Psi_i(a))\Big)$$

□

In the base case, the necessity degree of each atomic predicate is considered. The necessity degree of the ground terms true and false are, respectively, 1 and 0. For the elementary term $\Omega_{(i,j)}(a)$ forming $\Phi(a)$, we have several cases:

• When we have equal weights of all the criteria, then the weights are omitted in determining $N(\Omega(a))$.

• When both the certainty of a mapped to its sort is below 1 minus the weight assigned to the mapping, and the certainty of the relator $R^{\mathcal{D}}_{(i,j)}(a)$ used in Ω is also below 1 minus the weight assigned to the relationship, then the weights are also omitted. Hence, when $(SV^{\mathcal{D}}(a) \leq (1 - w_{\texttt{inst}})$ and $R^{\mathcal{D}}_{(i,j)}(a) \leq (1 - w_{\texttt{Wrel}})$ means that the importance or influence of the mapping of instances to sorts and the relator in the overall-uncertainty determination outweighs the level of uncertainty associated with it. That is why we ignore the weights in this case.

We can extend the necessity degree function to the datascape concepts as follows: $N(C_d) \overset{\text{def}}{=} N(\Phi) = \texttt{max}\Big(a \mid a \in A : N(\Phi(a))\Big)$, where $C_d = \{a \mid \tau(a) \in L \wedge \Phi(a)\}$ is a datascape concept that is defined within a DIS $\mathcal{D}$. We take the max of the individual necessity degrees due to the union property described earlier in Equation 2.

For objective concepts in the lattice, the composition operator $\oplus$ enables the formation of new concepts by composing existing ones i.e., creating composite concepts from the set of atomic concepts T. Writing $k = k_1 \oplus k_2$ means that concept k is constructed by the Cartesian product of concepts k_1 and k_2. These concepts are certain and carry no uncertainty. An alternative way to consider uncertainty in an objective concept is considering its specializing predicate that is always true, hence its certainty degree is 1.

4 REASONING ON POSSIBILISTIC DIS FRAMEWORK

We discuss several reasoning tasks and their governing inference rules for deriving conclusions in different reasoning scenarios. These tasks are concept satisfiability and concept subsumption. Each of which is explained in detail. We use N to denote the necessity degree function.

4.1 Concept Satisfiability

In this subsection, we examine concept satisfiability in necessity-based reasoning within the DIS framework, distinguishing between the objective and the datascape concept satisfiability.

In the classical DIS framework, a datascape concept is considered satisfiable if its corresponding data values exist within the carrier set of the DDV. However, in the necessity-based extension of DIS, we introduce the necessity degree to account for incomplete information of the data specializing predicate ($\Phi(a)$), which defines the datascape concept. In this extended framework, a datascape concept C_d is deemed satisfiable if there exists at least one instance $a \in C_d$ such that the necessity degree $N(a, C_d)$ of this instance is strictly greater than zero. Therefore, a datascape concept is satisfiable if and only if its necessity degree is strictly greater than zero, indicating that there is sufficient data support for the concept's existence.

Definition 3. *Let $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$ be a given DIS. Let $C_d = \{a \mid \tau(a) \in L \wedge \Phi(a)\}$ be a datascape concept that is defined within $\mathcal{D}$, with $\Phi(a)$ as its data specializing predicate. The datascape concept C_d is satisfiable, denoted by $\texttt{stsfd}(C_d)$, if and only if $\exists(a \mid a \in A : N(\Phi(a)) > 0)$.*

For objective concepts within the Boolean lattice, their certainty is inherently guaranteed, as they are directly linked to the DDV of the DIS under consideration. Thus, their satisfiability is inherently ensured, meaning they are both valid and certain to exist. The satisfiability of a composite concept is also guaranteed, as its atomic components have a degree of necessity of one. In this case, the combination of their necessity degree results in the composite concept also having a necessity degree of one, ensuring its satisfiability. If, from another perspective, one sees objective concepts as concepts that are independent of datasets, which translates into a data specializing predicate equivalent to true, then using Definition 3 and Procedure 3.1(item 1), one infers that its necessity degree is also equal to one.

Claim 4.1. *Let C_{d1} and C_{d2} be datascape concepts defined in a given DIS. Let $\Phi_1(a)$ and $\Phi_2(a)$ be their data specializing predicates, respectively. We have*

$$\texttt{stsfd}(C_{d1} \oplus C_{d2}) \equiv \texttt{stsfd}(C_{d1}) \vee \texttt{stsfd}(C_{d2}).$$

Proof. The concepts C_{d1} and C_{d2} are two datascape concepts. Hence, by Definition 1 and for $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$ is a given DIS, we can write C_{d1} and C_{d2} as follows: $C_{d1} \overset{\text{def}}{=} \{a \mid \tau(a) \in L \wedge \Phi_1(a)\}$, and $C_{d2} \overset{\text{def}}{=} \{a \mid \tau(a) \in L \wedge \Phi_2(a)\}$.

Then, we have $\texttt{stsfd}(C_{d1} \oplus C_{d2})$

$\equiv \quad \langle$ Definition 2 $\rangle$

$\texttt{stsfd}\{a \mid \tau(a) \in L \wedge (\Phi_1(a) \ \vee \ \Phi_2(a))\}$

$\equiv$ $\langle$ Definition 3: Satisfiability of datascape concept $\rangle$

$\exists(a \mid a \in A : N(\Phi_1(a)) > 0 \ \vee \ N(\Phi_2(a)) > 0\,)$

$\equiv$ $\langle$ Axiom Distributivity for $\vee$ $\rangle$

$\exists(a \mid a \in A : N(\Phi_1(a)) > 0\,)$

$\vee\ \exists(a \mid a \in A : N(\Phi_2(a)) > 0\,)$

$\equiv$ $\langle$ Definition 3 $\rangle$

$\mathtt{stsfd}(C_{d1}) \ \vee \ \mathtt{stsfd}(C_{d2})$

□

Example 4.1 (Satisfiability of a Datascape Concept). *Consider the datascape concept* $\mathtt{PositiveFeedback} = \{a \mid \tau(a) \in \mathtt{CustomerService} \wedge a.\mathtt{Satisfaction} \geq 0.6\}$. *Thus, this concept consists of a single atomic predicate:* $\Omega(a) = (a.\mathtt{Satisfaction} \geq 0.6)$. *Assume the following information is provided by a domain expert:*

- *Instance distribution relation:*
$$SV^{\mathcal{D}}(a_1) = (\mathtt{Good}, (\mathtt{Satisfaction}, 0.7))$$
- *Relator necessity degree:* $R^{\mathcal{D}}(a_1) = (\mathtt{Good} \geq 0.6,\ 0.8)$
- *Wights of importance:* $w_{\mathtt{inst}} = 0.4, w_{\mathtt{Wrel}} = 0.6$

Then, using Procedure 3.1, we compute the necessity degree of the atomic predicate:

$$\mathtt{min}\Big(\mathtt{max}(1 - w_{\mathtt{inst}}, SV^{\mathcal{D}}(a_1)),\ \mathtt{max}(1 - w_{\mathtt{Wrel}}, R^{\mathcal{D}}(a_1))\Big)$$
$$= \mathtt{min}(\mathtt{max}(0.6, 0.7),\ \mathtt{max}(0.4, 0.8)) = 0.7$$

Since $\Phi(a)$ *consists of just this atomic predicate, we have:*

$$N(C_d) = N(\Phi(a)) = N(\Omega(a_1)) = 0.7$$

By Definition 3, the concept is satisfiable because $N(\Phi(a)) > 0$. *Hence:* $\mathtt{stsfd(PositiveFeedback)}$ *holds.* □

4.2 Necessity-Based Subsumption

In general, we say that a concept C_1 subsumes a concept C_2 if every instance of C_2 is in C_1. In classical DIS, we have an additional kind of subsumption relationship. It is the `partOf` relationship, denoted by $\sqsubseteq_c$, that exists among the members of the Boolean lattice. When we write $C_1 \sqsubseteq_c C_2$, it indicates that the instances of C_1 are obtained through the projection of corresponding instances of C_2 on the attributes defining C_1. In this case, we say that C_2 subsumes C_1. The formal definition of DIS based subsumption is given below:

Definition 4. *Given a DIS* $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$. *Let* C_1 *and* C_2 *be two concepts in the set of concepts of* $\mathcal{D}$. *We say that* $C_2 \sqsubseteq C_1$ *iff one of the conditions holds:*

1. $C_1 \in L \ \wedge\ C_2 \in L \ \wedge\ C_2 \sqsubseteq_c C_1$.

2. C_1 *and* C_2 *are two datascape concepts with data specializing predicates* Φ_1 *and* Φ_2, *respectively and* $\forall(a \mid a \in A : \Phi_2(a) \implies \Phi_1(a)\,)$.

3. $C_1 \in L$ *and* C_2 *is a datascape concept. We have* $(C_1, C_2) \in R^*$, *where* R^* *is the reflexive transitive closure of a relation* R *of the graph rooted at* C_1.

Definition 4 formalizes concept subsumption in DIS, covering both objective and datascape concepts. First, if C_1 and C_2 are objective concepts in the ontology lattice, subsumption holds if $C_2 \sqsubseteq_c C_1$, meaning C_2 is structurally more specific than C_1 per the lattice order, reflecting the traditional subclass relation of the lattice hierarchical structure. Second, if both are datascape concepts defined by data specializing predicates Φ_1 and Φ_2, then $C_2 \sqsubseteq C_1$ holds if $\forall a \in A$, the implication $\Phi_2 \implies \Phi_1$ is satisfied. This ensures all instances satisfying C_2 also satisfy C_1. Third, if C_1 is an objective concept and C_2 is a datascape, subsumption holds if a path exists from C_1 to C_2 in the reflexive-transitive closure R^* of relation R. Notably, all concepts in the graph rooted at C_1 are considered a specialization of C_1. Subsumption is a partial order (reflexive, antisymmetric, transitive) over concepts, as stated in the following claim.

Claim 4.2. *Given a DIS* $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$, *the subsumption relation on the set of concepts in* $\mathcal{D}$ *is a partial order.*

Proof. We provide a proof for each case of the subsumption relation as given in Definition 4.

1. In the first case, the subsumption relation is identical to the `partOf` (i.e., $\sqsubseteq_c$) relation. The latter satisfies the properties required for subsumption (reflexivity, transitivity, and anti-symmetry) because it is defined on a Boolean lattice, which is itself a partially ordered set (poset) (Marinache, 2025).

2. In the second case, the subsumption relation $\sqsubseteq$ defines a partial order over the datascape concepts. This is due to the properties of the logical $\implies$ operation that satisfies the properties of partial order (Gries and Schneider, 1993, Pages 57-59).

3. In the third case, subsumption is interpreted as membership to the reflexive transitive closure R^*. It is established that a reflexive transitive closure on an acyclic graph is a partial order. Indeed, the rooted graphs are acyclic, as furthermore each is a Directed Acyclic Graph (DAG).

□

In the context of necessity-based subsumption, the subsumption necessity degree is determined by a domain expert. In addition, as the transitivity of subsumption applies, the degree of transitive subsumption is governed by the weakest link resolution rule, presented previously in subsection 2.2. This approach to possibilistic transitivity reasoning has been adopted in prior research (e.g., (Mohamed et al., 2018; Benferhat and Bouraoui, 2015)) and has shown effectiveness in handling uncertainty within possibilistic ontologies.

Claim 4.3. *Let $\mathcal{D} = (\mathcal{O}, \mathcal{A}, \tau)$ be a DIS. Let C_1, C_2, and C_3 be concepts defined in $\mathcal{D}$. Let*

$$R_{\sqsubseteq} = \{(C_1, C_2) \mid C_1 \in C \wedge C_2 \in C \wedge C_1 \sqsubseteq C_2\}$$

and $R^{\mathcal{D}}_{\sqsubseteq}$ is its corresponding necessity relation. We have:

$$((C_1,C_2),\alpha) \in R^{\mathcal{D}}_{\sqsubseteq} \wedge ((C_2,C_3),\beta) \in R^{\mathcal{D}}_{\sqsubseteq} \implies ((C_1,C_3), min(\alpha,\beta)) \in R^{\mathcal{D}}_{\sqsubseteq}$$

Proof. $((C_1,C_2),\alpha) \in R^{\mathcal{D}}_{\sqsubseteq} \wedge ((C_2,C_3),\beta) \in R^{\mathcal{D}}_{\sqsubseteq}$

$\equiv$ $\langle$ Definition of $R^{\mathcal{D}}_{\sqsubseteq}$ $\rangle$

$(C_1,C_2) \in R^{\mathcal{D}}_{\sqsubseteq} \wedge (C_2,C_3) \in R^{\mathcal{D}}_{\sqsubseteq}$
$\wedge N((C_1,C_2)) = \alpha \wedge N((C_2,C_3)) = \beta$

$\implies$ $\langle$ Transitivity of $R_{\sqsubseteq}$ and the weakest link resolution rule (Equation 3) $\rangle$

$(C_1,C_3) \in R^{\mathcal{D}}_{\sqsubseteq} \wedge N((C_1,C_3)) = \mathtt{min}(\alpha,\beta)$

$\equiv$ $\langle$ Definition of $R^{\mathcal{D}}_{\sqsubseteq}$ $\rangle$

$((C_1,C_3), \mathtt{min}(\alpha,\beta)) \in R^{\mathcal{D}}_{\sqsubseteq}$

□

The necessity-based subsumption between objective concepts (`partOf` relation) invariably assumes a necessity degree of 1, since the parthood relation among lattice concepts is considered fully certain i.e., $N(\mathtt{partOf}) = 1$, as indicated previously in subsection 3.3. This certainty extends naturally to the transitivity of `partOf` relation, whereby the minimum necessity degree computed over a chain of parthood relations, among lattice concepts, gives a degree of 1.

Example 4.2 (Transitivity of Concept Subsumption in DIS)**.** *Let $C_1 = \mathtt{Complaints}$, $C_2 = \mathtt{Feedback}$, $C_3 = \mathtt{CustomerService}$ be three concepts defined in given DIS. Suppose the following necessity-based subsumption relationships are provided by the domain expert:*

$$\{((\mathtt{Complaints}, \mathtt{Feedback}), 0.6), ((\mathtt{Feedback}, \mathtt{CustomerService}), 0.8)\} \subseteq R^{\mathcal{D}}_{\sqsubseteq}$$

By Claim 4.3, the transitive subsumption relation holds with:

$$N(\mathtt{Complaints}, \mathtt{CustomerService}) = 0.6$$

5 RELATED WORK AND DISCUSSION

This section reviews ontology modelling approaches that handle uncertainty using possibility theory, and explains our choice of this formalism.

In possibilistic DL-based approaches, uncertainty is modelled by assigning necessity or possibility degrees to ontology axioms at different levels. For instance, Pb-π-DL-Lite (Boutouhami et al., 2017) assigns necessity degrees only to ABox assertions, allowing uncertain instance membership such as $(\mathtt{Status}, \mathtt{Good}) = 0.9$ in the `CustomerService` domain, without modelling uncertainty at concept or relationship levels. The work of (Sun, 2013) focuses on uncertainty at the TBox level, assigning necessity degrees to TBox axioms such as $N(\mathtt{isA}(\mathtt{Rating}, \mathtt{Feedback})) = 0.8$, yet does not support uncertain instance classification or data-driven concept definitions. Similarly, studies such as (Benferhat and Bouraoui, 2015; Benferhat et al., 2014; Qi et al., 2011) extend possibilistic logic to both TBox and ABox axioms, enriching the expressiveness by allowing weighted axioms at multiple levels; however, their frameworks still assume that concepts like `PositiveFeedback` are defined and do not enable concept definitions directly derived from data (i.e., datascape concepts). The study of (Mohamed et al., 2018) further incorporates possibility distributions over interpretations, adding expressiveness to represent uncertainty about models themselves, but does not provide mechanisms to ground concepts in uncertain data attributes or integrate graded uncertainty at the attribute-concept mapping level.

At the language level, (Safia and Aicha, 2014) propose extending Web Ontology Language 2 (OWL2) with possibilistic annotations, enabling uncertainty representation in both concepts and instances. Using the `CustomerService` example, one could annotate a concept like `PositiveFeedback`, or an instances like `Fast` with possibility degrees, yet the approach still requires concepts to be pre-defined and does not support automatic or context-aware construction of concepts from data conditions. Finally, the work of (Ben Salem et al., 2018) assigns possibility degrees directly to concepts outside DL semantics, like assigning possibility degree to the concept `PositiveFeedback`, but does not address uncertainty propagation from attribute data to instances or model uncertainty in relationships or attribute mappings.

In contrast, our DIS-based framework uniquely integrates uncertainty at all levels: attributes, instances, relationships, and data-driven concepts. For example, we model uncertain attribute-concept mappings such as $N(\texttt{Quality} \rightarrow \texttt{Status}) = 0.7$, uncertain instance-concept classification like $(\texttt{Good}, 0.9)$, and relationships such as $N(\texttt{isA}(\texttt{Rating}, \texttt{Feedback})) = 0.9$. Our datascape concept `PositiveFeedback` is defined by a data-specializing predicate reflecting the actual satisfaction values, allowing uncertainty in satisfaction data to propagate naturally to the membership degree in `PositiveFeedback` concept. This unified, data-grounded modelling allows more expressive, context-aware reasoning about uncertain information compared to existing possibilistic ontology methods.

From a methodological standpoint, choosing an uncertainty formalism requires alignment with the modelling goals and constraints of the framework. While several candidates exist, including probabilistic approaches and Dempster-Shafer Theory (DST), we adopt possibility theory for its suitability to our framework. Probability theory enforces the additivity axiom, requiring the sum of probabilities for mutually exclusive events in a universe of discourse to equal one even under insufficient data (Kovalerchuk, 2017), leading to challenges in accurately representing uncertainty. In contrast, possibility theory relaxes this constraint, making it more suitable for the proposed approach. This rationale is supported by several possibilistic ontology frameworks (e.g., (Bal-Bourai and Mokhtari, 2016; Boutouhami et al., 2017)). Regarding DST, it is primarily designed for belief fusion from multiple sources (McClean, 2003), whereas our approach derives certainty from a single source. Moreover, DST typically assigns belief to sets of hypotheses rather than individual ones, making it more suitable for representing group-level uncertainty. In contrast, the DIS framework demands fine-grained certainty assignments to individual attributes, values, and relationships (Sentz and Ferson, 2002; Gordon and Shortliffe, 1984). Possibility theory directly supports this by enabling necessity degrees to annotate specific elements, making it a natural fit for our ontology-based model.

6 CONCLUSION AND FUTURE WORK

This paper presents a principled extension of the DIS framework to support reasoning under incomplete information using necessity-based possibilistic logic. Unlike most ontology-based systems that assume complete information, our approach models uncertainty across instances, attributes, relationships, and concept definitions, enabling fine-grained, graded reasoning. A key advantage is replacing binary inferences with necessity-valued conclusions, allowing cautious reasoning with partial information. Overall, this approach provides a structured foundation for possibilistic reasoning in ontology-based systems advancing more expressive and uncertainty-aware knowledge representations essential for robust decision-making in complex, data-limited contexts.

We are currently automating necessity-based reasoning tasks using the Domain Information System Extended Language (DISEL) tool (Wang et al., 2022). Future work will focus on automating necessity degree assignment via machine learning, integrating fuzzy logic to handle imprecision, developing a scalable reasoning engine, and applying the framework to real-world domains. Once the automation is in place, we plan to use DISEL to reason over data collected from network security prevention mechanisms. This data is often uncertain and originates from diverse sources with varying levels of reliability. Furthermore, this data originates from log files, whether structured or semi-structured, making it well-suited for DIS modelling. The goal is to preprocess and clean the data (Khedri et al., 2013), then apply the proposed reasoning framework to facilitate reliable and context-aware security decision-making in highly dynamic and complex uncertain landscapes.

REFERENCES

Alola, A., Tunay, M., and Alola, U. (2013). Analysis of possibility theory for reasoning under uncertainty. *International Journal of Statistics and Probability*, 2:12.

Alomair, D. and Khedri, R. (2025). Towards a cartesian theory of ontology domain adequacy. *Synthese*, pages 1–12. Submitted for publication on July. 25, 2025.

Alomair, D., Khedri, R., and MacCaull, W. (2025). A comprehensive review of uncertainty modelling in domain ontologies. *ACM Computing Surveys*, pages 1–35. Accepted with minor revision.

Anand, S. K. and Kumar, S. (2022). Uncertainty analysis in ontology-based knowledge representation. *New Generation Computing*, 40(1):339–376.

Bal-Bourai, S. and Mokhtari, A. (2016). Poss-SROIQ(D): Possibilistic description logic for uncertain geographic information. In *Trends in Applied Knowledge-Based Systems and Data Science*, pages 818–829, Cham. Springer International Publishing.

Ben Salem, Y., Idoudi, R., Ettabaa, K. S., Hamrouni, K., and Soleiman, B. (2018). Mammographie image based possibilistic ontological representation. In *(ATSIP) 4th International Conference*, pages 1–6.

Benferhat, S. and Bouraoui, Z. (2015). Min-based possibilistic DL-Lite. *Journal of Logic and Computation*, 27(1):261–297.

Benferhat, S., Bouraoui, Z., and Tabia, K. (2014). On the revision of prioritized DL-Lite knowledge bases. In *Scalable Uncertainty Management*, pages 22–36, Cham. Springer International Publishing.

Bosc, P. and Prade, H. (1997). *An Introduction to the Fuzzy Set and Possibility Theory-Based Treatment of Flexible Queries and Uncertain or Imprecise Databases*, pages 285–324. Springer US, Boston, MA.

Boutouhami, K., Benferhat, S., Khellaf, F., and Nouioua, F. (2017). Uncertain lightweight ontologies in a product-based possibility theory framework. *International Journal of Approximate Reasoning*, 88:237–258.

Ceravolo, P., Damiani, E., and Leida, M. (2008). Which role for an ontology of uncertainty? In *International Workshop on Uncertainty Reasoning for the Semantic Web*, volume 423.

Dubois, D., Lang, J., and Prade, H. (1994). Possibilistic logic. *Handbook of logic in artificial intelligence and logic programming*, pages 439–513.

Dubois, D. and Prade, H. (1986). Weighted minimum and maximum operations in fuzzy set theory. *Information Sciences*, 39(2):205–210.

Dubois, D. and Prade, H. (2014). Possibilistic logic—an overview. *Handbook of the History of Logic*, 9:283–342.

Dubois, D. and Prade, H. (2015). Possibility theory and its applications: Where do we stand? *Mathware and Soft Computing Magazine*, 18.

Gordon, J. and Shortliffe, E. H. (1984). The dempster-shafer theory of evidence. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, 3(832-838):3–4.

Grabisch, M. (1998). *Fuzzy Integral as a Flexible and Interpretable Tool of Aggregation*, pages 51–72. Physica-Verlag HD, Heidelberg.

Gries, D. and Schneider, F. B. (1993). *A logical approach to discrete math*. Springer-Verlag, Berlin, Heidelberg.

Imieliński, T. and Lipski, W. (1984). The relational model of data and cylindric algebras. *Journal of Computer and System Sciences*, 28(1):80–102.

Khedri, R., Chiang, F., and Sabri, K. E. (2013). An algebraic approach for data cleansing. In *the 4th EUSPN*, volume 21 of *Procedia Computer Science*, pages 50 – 59. Procedia Computer Science.

Kovalerchuk, B. (2017). *Relationships Between Probability and Possibility Theories*, pages 97–122. Springer International Publishing, Cham.

Laha, R. G. and Rohatgi, V. K. (2020). *Probability theory*. Courier Dover Publications.

Ma, Z. M., Zhang, F., Wang, H., and Yan, L. (2013). An overview of fuzzy description logics for the semantic web. *The Knowledge Engineering Review*, 28(1):1–34.

Marinache, A. (2025). *Syntax and Semantics of Domain Information System and its usage in conjecture verification*. PhD thesis, School of Graduate Studies, McMaster University, Hamilton, Ontario, Canada.

Marinache, A., Khedri, R., LeClair, A., and MacCaull, W. (2021). DIS: A data-centred knowledge representation formalism. In *(RDAAPS) 2021: A Big Data Challenge*, pages 1–8.

McClean, S. I. (2003). Data mining and knowledge discovery. In *Encyclopedia of Physical Science and Technology (Third Edition)*, pages 229–246. Academic Press, New York, third edition edition.

Mohamed, R., Loukil, Z., and Bouraoui, Z. (2018). Qualitative-based possibilistic el ontology. In *PRIMA 2018: Principles and Practice of Multi-Agent Systems*, pages 552–559, Cham. Springer International Publishing.

Nieves, J. C., Osorio, M., and Cortés, U. (2007). Semantics for possibilistic disjunctive programs. In *LPNMR: 9th International Conference, Tempe, AZ, USA, May 15-17, 2007. Proceedings 9*, pages 315–320. Springer.

Qi, G., Ji, Q., Pan, J. Z., and Du, J. (2011). Extending description logics with uncertainty reasoning in possibilistic logic. *International Journal of Intelligent Systems*, 26(4):353–381.

Safia, B.-B. and Aicha, M. (2014). Poss-OWL 2: Possibilistic extension of OWL 2 for an uncertain geographic ontology. *Procedia Computer Science*, 35:407–416.

Sentz, K. and Ferson, S. (2002). *Combination of evidence in Dempster-Shafer theory*, volume 4015. Sandia National Laboratories Albuquerque.

Straccia, U. (2013). *Foundations of Fuzzy Logic and Semantic Web Languages*. Chapman & Hall/CRC.

Straccia, U. and Bobillo, F. (2017). *From Fuzzy to Annotated Semantic Web Languages*, pages 203–240. Springer International Publishing, Cham.

Sun, S. (2013). A novel semantic quantitative description method based on possibilistic logic. *Journal of Intelligent & Fuzzy Systems*, 25:931–940.

Wang, Y., Chen, Y., Alomair, D., and Khedri, R. (2022). DISEL: A language for specifying DIS-based ontologies. In *(KSEM) 2022*, Lecture Notes in Artificial Intelligence, pages 1–16, Singapore. Springer.

Zadeh, L. (1999). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100:9–34.