

Conflict-Based Inconsistency-Tolerant Query Answering

Rim Mohamed and Zied Loukil and Faiez Gargouri

rymmohammed2@gmail.com

zied.loukil@isims.usf.tn

faiez.gargouri@usf.tn

University of Sfax, Miracl Laboratory, Isims, Tunisia

Zied Bouraoui

bouraoui@cril.fr

CRIL CNRS & Univ Artois, Lens, France

Abstract

In this paper, we place ourselves in the context of inconsistency-tolerant query answering over lightweight ontologies, which aims to query a set of conflicting facts using an ontology that represents generic knowledge about a particular domain. Existing inconsistency-tolerant semantics typically consist in selecting some of (maximal) consistent subsets of facts, called repairs. We explore a novel strategy to select the most relevant repairs based on the stratification of the assertional base into priority levels that we automatically induce from the ontology. We propose a method that exploits conflict statistical regularities between facts to induce an embedding, in which each fact is represented by a vector. Based on Euclidean distances between facts, we classify the assertions from the most reliable to the least important ones. We then use these distances to define relevant repairs. Interestingly enough, we show that the obtained repair is done in polynomial time.

Introduction

Ontology-mediated query answering (OMQA) provides query reformulation techniques over ontological domain knowledge to improve access to data (Bienvenu 2020). Recent years have witnessed an increased interest in using *OWL2-QL* (Lemos 2019) profile for OMQA, which is based on a family of lightweight description logic, called DL-Lite (Calvanese et al. 2005). DL-Lite is designed for applications that use huge data where query answering is the most important task. While in OMQA setting the ontological knowledge is assumed to be satisfiable, fully reliable, and often debugged by experts, the data (i.e., the assertional base), however, are usually of bad quality. This for example may happen when collecting data from several sources (Benferhat, Bouraoui, and Tabia 2015), or due to the ontology mapping (Mani and Annadurai 2021) or ontology evolution (Algo-saibi 2021). When the data are conflicting with the ontology, logical deduction used for query answering is no longer appropriate, i.e, every fact can be derived as an answer to a query including the conflicting facts causing the inconsistency (ex falso quodlibet sequitur). For example, if an ontology indicates that it is not possible to be a professor and a

student at the same time and that an individual John is said to be both, then the knowledge base will allow us to imply not only that John is a professor and a student, but also that John is a teaching module, which is undesirable.

This observation has led to several works on handling inconsistency in OMQA where several inconsistency-tolerant inference relations, called semantics, have been proposed (Bienvenu 2020). Most of these semantics, inspired by database reparation (Bertossi 2019) or nonmonotonic reasoning in propositional logic, consist in getting rid of inconsistency by first computing a set of (maximally) consistent subsets of the assertional base, called repairs, and then using them to perform query answering. For example, the AR semantics (Lembo et al. 2010; Belabbes, Benferhat, and Chomicki 2021) consists in computing all the inclusion-maximal subsets of data that are consistent with the ontology and considering an answer as valid if it holds all the repairs. In (Baget et al. 2016), a general framework that unifies inconsistency-tolerant semantics is defined. The idea behind our framework is to distinguish between the way data assertions are virtually distributed (the notion of modifiers) and inference strategies. An inconsistency-tolerant semantics is then naturally defined by a modifier and an inference strategy.

In some scenarios, the data are coming from different conflicting sources having different reliability levels. To take into account this information while handling inconsistency, several inconsistency-tolerant semantics have been considered based on the notion of preferred repairs when the assertions base is prioritized (Bourgau 2016). When there exists a total pre-order between assertions, the non-defeated repair semantics (Benferhat, Bouraoui, and Tabia 2015) is defined. The idea of this semantics consists of retrieving from each stratum only the set of free assertions to obtain a single repair. The importance of this repair semantics is that it has been done in polynomial time. Preferred repair semantics (Bienvenu 2020) is also designed for the same case as non-defeated repair semantics (i.e., the total preorders). In this area, they investigated variants of the AR (Lembo et al. 2010), and IAR (Lembo et al. 2015) semantics by changing the classical notion of repair by the different types of preferred repairs, namely cardinality, prioritized set inclusion, prioritized cardinality, and weights.

In this paper, we follow another direction for

inconsistency-tolerant query answering by exploiting co-occurrence conflicting relations for selecting preferred repairs. We explore novel strategies to compute relevant repairs based on the stratification of the assertional base into priority levels that are automatically induced from the ontology by exploiting statistical analysis between conflicting facts to induce an embedding, in which each fact is represented by a vector. We then propose a method that identifies repairs based on the Euclidean distance between facts and define a new repairing strategy that first classifies the facts into a set of clusters and then uses the Euclidean distance between their centroids to select one repair. We show, in particular, that all obtained repairs are tractable.

Background

In this section, we briefly recall the syntax and semantics of DL-Lite, which underlies *OWL2-QL* ontology language designed for applications that use huge volumes of data and review the main existing inconsistency-tolerant semantics proposed to deal with inconsistency.

DL-Lite syntax and semantics. Let N_C, N_P, N_I be three pairwise disjoint sets where N_C denotes a set of atomic concepts, N_P denotes a set of atomic roles and N_I denotes a set of individuals. The DL-Lite concept expressions are built according to the following syntax:

$$\begin{aligned} R &\longrightarrow P|P^- & E &\longrightarrow R|\neg R \\ B &\longrightarrow A|\exists R & C &\longrightarrow B|\neg B \end{aligned}$$

Where A is an atomic concept, P is an atomic role and P^- is the role inverse of P . B and C are basic and complex concepts. Finally, the role R (resp E) is basic (resp. complex) role. According to the above expression, the DL-Lite logic uses three unary connectors, the negation (\neg), the existence (\exists) and the inverse ($-$).

The DL-lite knowledge base $\mathcal{K} = (\mathcal{A}, \mathcal{T})$ is composed by a set of assertions (\mathcal{A}) on atomic concepts and on atomic roles of the form $A(a)$ and $P(a, b)$ respectively. The terminological part (\mathcal{T}) is composed by a finite set of inclusion axioms between concepts of the form $A \sqsubseteq B$.

The semantics is given in terms of interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ which consist of a non-empty interpretation domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ that maps each individual $a^{\mathcal{I}} \in N_I$ to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each concept $A \in N_C$ to a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each role $P \in N_P$ to a subset $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The interpretation function $\cdot^{\mathcal{I}}$ is extended in a straightforward way for complex concepts and roles, i.e., $(P^-)^{\mathcal{I}} = \{(y, x) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} | (x, y) \in P^{\mathcal{I}}\}$, $(\exists R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} | \exists y \in \Delta^{\mathcal{I}} \text{ such that } (x, y) \in R^{\mathcal{I}}\}$, $(\neg B)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus B^{\mathcal{I}}$, $(\neg R)^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}$.

An interpretation \mathcal{I} is said to be a model of (or satisfies) a GCI, denoted by $\mathcal{I} \models C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Similarly, \mathcal{I} satisfies a concept (resp. role) assertions, denoted $\mathcal{I} \models C(a)$ (resp. $\mathcal{I} \models P(a, b)$), if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$). A knowledge base $\mathcal{K} = (\mathcal{A}, \mathcal{T})$ is said to be consistent if it has a model. Otherwise, it is inconsistent. An axiom ϕ is entailed by \mathcal{K} , denoted by $\mathcal{K} \models \phi$, if ϕ is satisfied by every model of \mathcal{K} .

Inconsistency-Tolerant Semantics Several inconsistency-tolerant semantics have been proposed in the literature to deal with inconsistency in OMQA and obtain meaningful answers to queries. Most of these semantics are based on the notion of repair that can be used to perform inference, i.e., query answering.

Definition 1 (A Repair). A repair, denoted by \mathcal{R} , is an inclusion-maximal subset of the ABox which is consistent with the TBox. More formally $(\mathcal{T}, \mathcal{R})$ is consistent iff $\forall \mathcal{R}' \subseteq \mathcal{A}$: if $\mathcal{R} \subsetneq \mathcal{R}'$ then $(\mathcal{T}, \mathcal{R}')$ is inconsistent.

Let us denote by $R(\mathcal{A})$ the set of repairs of ABox \mathcal{A} . Note that the inconsistency problem is always defined for some ABox, since a TBox may be incoherent but never inconsistent. Incoherence means that there is at least a model for \mathcal{T} , but for a concept C , such that $C^{\mathcal{I}} = \emptyset$. Inconsistency means there is a model. In DL-Lite, when the TBox is coherent, a conflict involves exactly two assertions (a_1, a_2) and removing one of them restores the consistency. In the following, we recall the most common inconsistency-tolerant semantics.

Definition 2 Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite ontology, the most common inconsistency-tolerant semantics are:

- **ABox Repair Semantics:** A tuple a is an answer to a query q under AR semantics, denoted $\mathcal{K} \models_{AR} q(a)$, if $\forall A' \in R(\mathcal{A}), \langle \mathcal{T}, A' \rangle \models q(a)$
- **Intersection of ABox Repair Semantics:** A tuple a is an answer to query q under IAR semantics, denoted $\mathcal{K} \models_{IAR} q(a)$, if $\langle \mathcal{T}, \mathcal{S} \rangle \models q(a)$, where $\mathcal{S} = \bigcap_{A' \in R(\mathcal{A})} A'$
- **Brave Semantics:** A tuple a is an answer to a query q under brave semantics, denoted $\mathcal{K} \models_{brave} q(a)$, if for some repair $A' \in R(\mathcal{A}), \langle \mathcal{T}, A' \rangle \models q(a)$

Example 1 Consider the following ontology:

Carnivore \sqsubseteq \neg Granivorous	Carnivore \sqsubseteq \neg Herbivores
\exists liveIn $^-$ \sqsubseteq AquaticRegion	\exists liveIn $^-$ \sqsubseteq Carnivore
Granivorous \sqsubseteq \neg Herbivore	Herbivore \sqsubseteq WildAnimals
WildAnimals \sqsubseteq \neg AqAnimals	Carnivore \sqsubseteq Animal

The ABox \mathcal{A} contains the following assertions:
 $\mathcal{A} = \{ \text{Carnivore}(\text{lion}), \text{Herbivores}(\text{lion}), \text{Granivorous}(\text{lion}), \text{Carnivore}(\text{monkey}), \text{WildAnimal}(\text{monkey}), \text{AquaticAnimal}(\text{cow}), \text{WildAnimal}(\text{cow}), \text{liveIn}(\text{fish}, \text{sea}) \}$

In Example 1 the ontology is inconsistent as the assertions $\text{Carnivore}(\text{lion})$ and $\text{Herbivore}(\text{lion})$ are contradicting w.r.t axiom $\text{Carnivore} \sqsubseteq \neg \text{Herbivore}$. If the ontology is inconsistent, then everything can be inferred from it. One way to tackle this problem is to select some repairs. For example: $\mathcal{R}_1 = \{ \text{Carnivore}(\text{monkey}), \text{WildAnimal}(\text{monkey}), \text{Herbivore}(\text{lion}), \text{WildAnimal}(\text{cow}) \}$.

The inconsistency-tolerant semantics introduced in Definition 2 are based on repairs computed using the initial ABox. However, one can also define the same semantics on repairs computed from a closed ABox (e.g. CAR and ICAR semantics) or using closed repairs instead of repairs themselves, i.e., ICR semantics (Lembo et al. 2010; 2015). One can also use lexicographical criterion instead

of inclusion to define repairs. Finally, one can also define repairs on prioritized ABox. A prioritized ABox $\mathcal{A} = A_1 \cup \dots \cup A_n$ is partitioned into n layers (also called strata), where each partition A_i contains the set of assertions having the same level of priority i , and they are considered as more reliable than the ones in the previous layer A_j with $j > i$. The presence of additional information in the reliability of the ABox facts, such as prioritized cardinality, prioritized set inclusion, and weights, leads us to use them to identify the preferred repair (Bienvenu, Bourgaux, and Goasdoue 2016).

Among the existing semantics, the IAR-semantics is the most cautious in terms of inference as it is only based on free assertions. Contrarily, credulous semantics is the most productive. However, it is often considered too adventurous as the set of conclusions may be inconsistent w.r.t the ontology. The AR-semantics has considered the safest semantics as an answer to a query is valid if it can be deduced from each repair. However, query answering within the AR-semantics is co-NP-complete even for simple DL-Lite languages such as DL-Lite_{core}. The main question addressed in this paper is how to compute relevant repairs in a tractable manner by exploiting additional information about conflicting facts.

Flexible Conflict Representation

When repairing conflicting ontologies, one important piece of information that needs to be considered is the participation of each assertion in the conflict and to what extent a fact is likely to be incompatible with each other. To provide an answer to this question, we exploit conflict regularities between facts by computing an embedding (a vector space representation) of the assertional base in which the distance between two facts reflects their compatibility. To represent the conflict between each pair of assertions, we use a one-hot matrix encoding of conflict, called conflict matrix. The conflict matrix takes the set of assertions as rows and columns. We assign '1' when there exists a conflict between the two assertions and '0' otherwise.

Definition 3 (Conflict Matrix). A conflict matrix \mathcal{M} is a square matrix that represents the relation of conflicts between assertions. It takes the set of assertions (a_1, a_2, \dots, a_n) in their rows and columns.

$$\forall a_i, a_j \in M, \text{ if } M_{a_i, a_j} = \begin{cases} 1 & \text{then } a_i, a_j \text{ are in conflict} \\ 0 & \text{then } a_i, a_j \text{ are consistent} \end{cases}$$

Example 2 Table 1 represents the conflict matrix of the inconsistent ontology in Example 1.

	C(l)	H(l)	G(l)	C(m)	WA(m)	WA(c)	AA(c)
C(l)	0	1	1	0	0	0	0
H(l)	1	0	1	0	0	0	0
G(l)	1	1	0	0	0	0	0
C(m)	0	0	0	0	0	0	0
WA(m)	0	0	0	0	0	0	0
WA(c)	0	0	0	0	0	0	1
AA(c)	0	0	0	0	0	1	0

Table 1: Conflict Matrix of Example 1

The conflict matrix obtained by Definition 3 is square and sparse. Unless the TBox is incoherent, the diagonal values are '0' since there is no incoherency. In real applications, the conflict matrix is huge and difficult to manipulate. For such reason, we need to reduce the dimensions of the matrix. The dimensionality reduction methods are classified into two categories. The first category consists in removing the redundant features and only keeping the most salient features in the data. Therefore, there is no modification in the set of features. The second one consists in expressing the existing features as a combination of new features. In this paper, we will use a method of the latter category using multidimensional scaling as a technique, which allows in turn it preserves much information when it transforms the data from the high dimensions to the low ones.

Multidimensional Scaling of Conflict Matrix. Multidimensional scaling (Cox and Cox 2008) is a set of statistical techniques used in the field of information visualization (Spence 2001) to explore similarities in data. It is a projection or a mapping from a high-dimensional data set in a low-dimensional space. The MDS takes as input a square, a symmetric matrix that indicates the relationships between items. Given N points (x_1, x_2, \dots, x_n) in high-dimensional space p , MDS represents these points in low-dimensional space $m < p$ by N new points (y_1, y_2, \dots, y_n) . These results can be presented by a matrix D having the Euclidean distances between each pair of points. The similarity and the Euclidean distance are two opposite notions, the lower the distance, the greater the similarity. In this paper, we start with the conflict matrix (\mathcal{M}) , which represents the relation of conflict between facts. We apply the MDS into \mathcal{M} to reduce the dimensions of their data and obtain a new matrix D that contains the Euclidean distances between each pair of assertions. As we will use the matrix D for repairing the assertional base, as intuitively, D encodes statistical information about conflict co-occurrence. The following definition introduces the notion of similarity between assertions that will be considered to capture conflict regularities.

Definition 4 Let \mathcal{E} be the low dimensional space that represents a set of assertions. We say that two assertions are similar if the two assertions have the same degree of conflict and $d(a_i, a_j) = 0$, where $d(a_i, a_j)$ is the Euclidean distance between the two assertions.

Note that the degree of conflict presents the occurrence of each fact in the minimal inconsistent subset. The notion of similarity in Definition 4 is given by the Euclidean distance: An Euclidean distance value of zero indicates no conflict between assertions and they are similar, while the larger the distance value, the more severe the conflict and the least similar the assertions. Figure 1 depicts a multidimensional data of Example 1 in the two-dimensional space. As one can see, this representation intuitively reflects conflicting relationships between assertions. Namely the assertions in the centre are the most similar. They have a Euclidean distance equal to 0, meaning that these assertions do not occur in any conflict, i.e., they are free assertions (the assertions that form the IAR-repair). Now, when we move further away from the centre, the Euclidean distance increases, and therefore the

	C(l)	H(l)	G(l)	C(m)	WA(m)	WA(c)	AA(c)
C(l)	0.000	0.588	0.237	0.237	0.2268	0.347	0.347
H(l)	0.588	0.000	0.413	0.413	0.698	0.273	0.571
G(l)	0.227	0.698	0.285	0.285	0.000	0.428	0.2564
C(m)	0.237	0.412	0.000	0.000	0.285	0.143	0.228
WA(m)	0.237	0.413	0.000	0.000	0.286	0.143	0.228
WA(c)	0.347	0.270	0.143	0.143	0.143	0.426	0.325
AA(c)	0.379	0.571	0.228	0.228	0.256	0.325	0.000

Table 2: Euclidean Distance Between Assertions

degree of conflict increases as well. Notice that the complexity of MDS used for computing pairs of conflict is the same as computing the pair of similarity (or dissimilarity) presented in (Morrison, Ross, and Chalmers 2003), which is done in $\mathcal{O}(n^2)$.

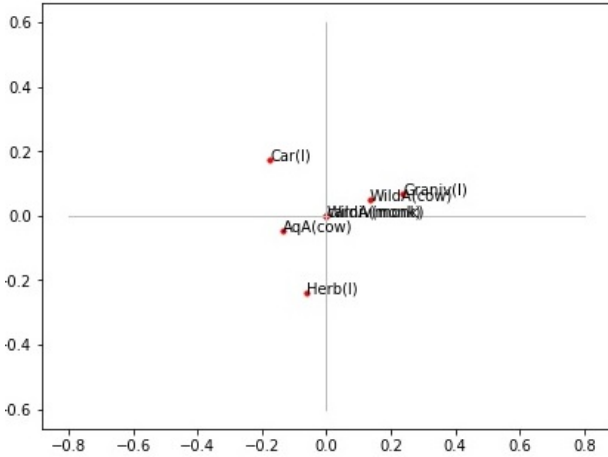


Figure 1: First two principal components of a vector space embedding of Example 1 using MDS.

Distance-Based Preferred Repairs

In this section, we will use MDS factorization of the conflict matrix to compute preferred repairs. Based on the Euclidean distances captured in an MDS conflicting space, we will induce a prioritized assertional base, starting with the most reliable facts until the set of the less important ones. More formally:

Definition 5 Let \mathcal{M} be a conflict matrix. Applying the MDS into \mathcal{M} leads to obtaining the Euclidean distance between each pair of assertions. Based on these distances, we define the stratified ABox as follows:

$$\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$$

Where \mathcal{A}_1 contains the most reliable assertions, those having Euclidean distance equal to zero (i.e., the free assertions). The assertions in \mathcal{A}_i are more reliable than those in the layer \mathcal{A}_j when $i < j$.

Recently, some strategies have been proposed to compute repairs of prioritized DL-Lite assertional bases (Benferhat et al. 2017), where the preorder between assertions is given

as input information (without justification). In this paper, we consider a pre-order that is automatically induced from the conflicting assertional base and use it to define relevant semantics based on the Euclidean distance between assertions (i.e., the distance between vectors in the space).

Min-Based Distance Repair

The min-based distance repair defined over the prioritized ABox (\mathcal{A}) with respect to the TBox (\mathcal{T}) is an inclusion-maximal subset of \mathcal{A} that is consistent with \mathcal{T} . The min-based distance repair selects the first layer, i.e., the free assertions, then, it iteratively selects from the entire set of facts those that are consistent and have the minimal distance w.r.t the existing ones. More formally:

Definition 6 Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ with $\mathcal{A} = \mathcal{A}_1, \dots, \mathcal{A}_n$ be the inconsistent DL-lite ontology. The min-based distance repair, denoted by Min_D , is defined as follows:

$$Min_D = \mathcal{A}_1 \cup \mathcal{A}'_2 \cup \dots \cup \mathcal{A}'_n$$

Where \mathcal{A}_1 contains the set of free assertions, and \mathcal{A}'_2 includes \mathcal{A}_1 and the consistent facts having the minimal distance w.r.t \mathcal{A}_1 .

It is important to note that we can have more than one Min_D repair of an ontology. Moreover, we can have two conflicting assertions at the same level, which are consistent with the existent ones. Therefore, there exist at least two repairs.

Example 3 Consider the inconsistent ontology in Example 1. First, The algorithm selects $Car(l)$ and $WA(l)$. Then, it finds two conflicting assertions in the same level $\{AA(c), WA(c)\}$. Therefore, we obtain two repairs $\mathcal{R}_1 = \{Car(l), WA(l), AA(c)\}$ and $\mathcal{R}_2 = \{Car(l), WA(l), WA(c)\}$.

The following proposition studies the computational complexity of Min_D

Proposition 1 Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent DL-Lite ontology. The min-based distance repair, denoted by Min_D is consistent and its computational complexity is in NP-complete.

The inconsistency-tolerant query answering under this semantics is the same as the AR-semantics which is co-NP-complete with respect to data complexity (Lembo et al. 2015). In the following, we will define another repair that is more cautious than the Min_D repair, but we can compute in polynomial time.

Central-Based Repair

The central-based repair is a consistent sub-base of the inconsistent ontology. This sub-base includes the set of free assertions (i.e., the assertions that do not involve any conflict). More formally:

Definition 7 Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, with $\mathcal{A} = \mathcal{A}_1 \cup \dots \cup \mathcal{A}_n$, be a stratified inconsistent DL-Lite ontology. The central repair, denoted by $CR(\mathcal{A})$, selects the set of consistent assertions of each layer, starting from \mathcal{A}_1 , and stops if it finds two conflicts assertions in the same level. More formally:

$$CR(\mathcal{A}) = \mathcal{A}_1 \cup \dots \cup \mathcal{A}_i$$

Where \mathcal{A}_{i+1} is inconsistent with \mathcal{A}_i .

The central repair is more cautious in terms of inference than the Min_D , as it is only based on assertions that are not conflicting with free assertions.

Example 4 *We continuous with the inconsistent ontology in Example 1. The $CR(\mathcal{A})$ is the intersection of Min_D repair : $CR(\mathcal{A}) = \{Carnivore(monkey), WildAnimal(monkey)\}$*

The following proposition shows that $CR(\mathcal{A})$ can be computed in polynomial time with respect to the size of the ABox.

Proposition 2 *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ with $\mathcal{A} = A_1 \cup A_2, \dots \cup A_n$. Let $CR(\mathcal{A})$ be its central repair. Then $CR(\mathcal{A})$ can be computed in polynomial time with respect to the size of the ABox.*

Proof 1 *The proof is given by the fact that computing the free sub-set is done in polynomial time.*

The CR-conclusions are considered safe since the central-based repair stops at layer A_i where there is a conflict assertion. Hence, only free assertions are taken into account for entailment. However, assertions having priority levels strictly greater than A_i are inhibited even if they are consistent. To overcome this problem and provide more productive repairs, we define centroid-based repair.

Centroid-Based Repair

This section presents new strategies for selecting one preferred repair that is more productive in terms of inference than the central repair. Selecting only one repair is important since it allows efficient query answering once the preferred repair is computed. These strategies are based on analyzing the minimal distance between centroids that can be computed from the matrix D . Only using Euclidean distances presented to compute the min-based distance repairs may lead to obtaining several repairs and in the worst case, the computational complexity is NP-complete. Computing Central repair is tractable, however, this repair is too cautious in terms of inference. We aim to define a new strategy of repair that provides a good compromise between productivity and computational complexity. To this end, the new strategy takes as input all the assertions and their Euclidean distances obtained by applying MDS.

To build centroid-based repair, we start with the assumption that each point (or assertion) is independent of the others and forms an individual cluster in the space. We then seek possible compatibilities between assertions, i.e., compatible clusters. If two assertions are similar, i.e., close in space and consistent with each other, then they are merged in the same cluster. After that, we obtain a set of clusters, each of which contains a set of consistent assertions. Let CL be the set of clusters, we compute for each $C \in CL$ the centroid as follows:

$$Cn = \left(\frac{\sum_{i=1}^{|C|} (x_i)}{|C|}, \frac{\sum_{i=1}^{|C|} (y_i)}{|C|} \right)$$

Where $\sum_{i=1}^{|C|} (x_i)$ is the sum of x_i of each point in the cluster C and $|C|$ is the cluster cardinality. After computing the centroid of each cluster, we compute the distance

between the centroid of the free cluster (the one that has the free assertions) and all the other centroids. The Euclidean distance between two centroids (m_1, m_2) having coordinates $(x_1, y_1), (x_2, y_2)$ respectively, is obtained by the following expression:

$$d(m_1, m_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

The next step consists of choosing the cluster that is more similar to the free one (i.e., the centroid having less Euclidean distance w.r.t the others). In the following, we define centroid-based repair.

Definition 8 *The centroid-based repair, denoted by $MinD_c$, is defined as follows: $MinD_c = C_1 \cup C'_2, \dots, \cup C'_n$, with:*

- C_1 is the free cluster, i.e., the cluster that contains the set of free assertions.
- $C'_i = C_1 \cup \min d(\text{free}(C_i))$ contains the set of consistent assertions of the cluster C_i that have the minimal distance w.r.t the others clusters.

In the first step, $MinD_c$ repair selects the free cluster and then chooses the cluster centroid that has the minimal distance w.r.t the free one. If there exist two centroids having the same distance w.r.t the free centroid, then we choose the cluster that has more consistent assertions. The following proposition studies the time complexity of the obtained repair and their consistency.

Proposition 3 *Let $MinD_c$ be the repair obtained by applying the centroid-based repair. Then, the $MinD_c$ is consistent and it is done in PTime.*

Comparative Study

In this section, we compare the repair strategies obtained in this paper based on their productivity of inference, i.e entailment of query answers. Before providing this study, we introduce the notion of deductive closure. In fact, the question-answering given in the previous section can be either defined on the assertional base \mathcal{A} and of its deductive closure $cl(\mathcal{A})$ defined as follows:

Definition 9 Deductive Closure *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite ontology. Let \mathcal{T}_p be the positive inclusion axioms of \mathcal{T} of the form $A \sqsubseteq B$ and \mathcal{T}_N be the set of negative inclusion axioms of the form $A \sqsubseteq \neg B$. The deductive closure of \mathcal{A} is defined as follows: $cl(\mathcal{A}) = \{B(a), \langle \mathcal{T}_p, \mathcal{A} \rangle \models B(a)\} \cup \{R(a, b), \langle \mathcal{T}_p, \mathcal{A} \rangle \models R(a, b)\}$, where B and R are a concept and a role of \mathcal{T} and a, b are individuals of \mathcal{A} .*

Note that the three algorithms presented above can be applied on $\langle \mathcal{T}, \mathcal{A} \rangle$ or $\langle \mathcal{T}, cl(\mathcal{A}) \rangle$. In the following we will extend the notion of deductive closure in case of prioritized ontology.

Definition 10 *let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, with $\mathcal{A} = \{A_1 \cup \dots \cup A_n\}$ be a DL-Lite ontology. We define the prioritized deductive closure of \mathcal{A} as follows: $cl(\mathcal{A}) = cl(A_1 \cup \dots \cup A_n)$.*

The following proposition shows that the Central repair is sensitive to the use of deductive closure.

Proposition 4 Let $CR(\mathcal{A})$ be the central repair of the inconsistent DL-Lite ontology and $CR(cl(\mathcal{A}))$ be central repair of $cl(\mathcal{A})$. Then $\forall q$:

- if $\langle \mathcal{T}, \mathcal{A} \rangle \models_{CR} q$ then $\langle \mathcal{T}, cl(\mathcal{A}) \rangle \models_{CR} q$. However the converse is false.

The Min_D and $MinD_c$ is also sensitive to the use of deductive closure. It is the same as $CR(\mathcal{A})$.

The following proposition shows that $CR(\mathcal{A}) \subseteq Min_D$ and $CR(\mathcal{A}) \subseteq MinD_c$

Proposition 5 Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an inconsistent DL-Lite ontology. Let $CR(\mathcal{A})$ be the central repair; Min_D be the min-based distance repair and $MinD_c$ be the centroid based repair. Then $CR(\mathcal{A}) \subseteq Min_D$ and $CR(\mathcal{A}) \subseteq MinD_c$

The following propositions study the productivity of the obtained repairs in terms of logical entailment.

Proposition 6 Each CR-consequence is also Min_D -consequence but the inverse is false.

The following proposition shows that each CR-consequence is also $MinD_c$ -consequence, however the inverse is false.

Proposition 7 Each CR-consequence is also $MinD_c$ -consequence but the inverse is false

Conclusion

In this paper, we represent novel strategies to compute repairs based on the stratification of the assertional base into priority levels that are automatically induced from the ontology. We propose a method that selects repairs based on the Euclidean distance between facts and define a new repairing strategy that first classifies the facts into a set of clusters and then use the Euclidean distance between their centroids to select one repair. We show that all obtained repairs are tractable. As future work, we plan consider other methods for inducing the prioritized ontology such as principal component analysis and singular value decomposition.

Acknowledgments

This work was supported by ANR CHAIRE IA BE4musIA and FEI INS2I 2022 EMILIE.

References

Algoasibi, A. A. 2021. Improving semantic properties relationships extraction in ontology evolution. *Journal of Theoretical and Applied Information Technology* 99(11).

Baget, J.-F.; Benferhat, S.; Bouraoui, Z.; Croitoru, M.; Mugnier, M.; Papini, O.; Rocher, S.; and Tabia, K. 2016. A general modifier-based framework for inconsistency-tolerant query answering. In Baral, C.; Delgrande, J. P.; and Wolter, F., eds., *KR 2016, Cape Town, South Africa, April 25-29, 2016*, 513–516. AAAI Press.

Belabbes, S.; Benferhat, S.; and Chomicki, J. 2021. Handling inconsistency in partially preordered ontologies: the elect method. *J. Log. Comput.* 31(5):1356–1388.

Benferhat, S.; Bouraoui, Z.; Papini, O.; and Wurbel, E. 2017. Prioritized assertional-based removed sets revision of dl-lite belief bases. *Annals of Mathematics and Artificial Intelligence*.

Benferhat, S.; Bouraoui, Z.; and Tabia, K. 2015. How to select one preferred assertional-based repair from inconsistent and prioritized dl-lite knowledge bases? In Yang, Q., and Wooldridge, M. J., eds., *Proceedings of the Twenty-Fourth International Joint Conference on AI, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 1450–1456. AAAI Press.

Bertossi, L. 2019. Database repairs and consistent query answering: Origins and further developments. In *Proceedings of the 38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, 48–58.

Bienvenu, M.; Bourgaux, C.; and Goasdoue, F. 2016. Explaining inconsistency-tolerant query answering over description logic knowledge bases. In AAAI.

Bienvenu, M. 2020. A short survey on inconsistency handling in ontology-mediated query answering. *Kunstliche Intell.* 34(4):443–451.

Bourgaux, C. 2016. *Inconsistency Handling in Ontology-Mediated Query Answering. (Gestion des incoherences pour l'accès aux données en présence d'ontologies)*. Ph.D. Dissertation, University of Paris-Saclay, France.

Calvanese, D.; Giacomo, G. D.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2005. Dl-lite: Tractable description logics for ontologies. In Veloso, M. M., and Kambhampati, S., eds., *Proceedings, Conference on AI, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, 602–607. AAAI Press / The MIT Press.

Cox, M. A., and Cox, T. F. 2008. Multidimensional scaling. In *Handbook of data visualization*. Springer. 315–347.

Lembo, D.; Lenzerini, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2010. Inconsistency-tolerant semantics for description logics. In Hitzler, P., and Lukasiewicz, T., eds., *Web Reasoning and Rule Systems - Fourth International Conference, RR 2010, Bressanone/Brixen, Italy, September 22-24, 2010. Proceedings*, volume 6333 of *Lecture Notes in Computer Science*, 103–117. Springer.

Lembo, D.; Lenzerini, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2015. Inconsistency-tolerant query answering in ontology-based data access. *Journal of Web Semantics*.

Lemos, J. C. L. 2019. *Tractable Ontology-Mediated Query Answering with Datatypes*. The University of Liverpool (United Kingdom).

Mani, S., and Annadurai, S. 2021. Explicit link discovery scheme optimized with ontology mapping using improved machine learning approach. *Studies in Informatics and Control* 30(1):67–75.

Morrison, A.; Ross, G.; and Chalmers, M. 2003. Fast multi-dimensional scaling through sampling, springs and interpolation. *Information Visualization*.

Spence, R. 2001. *Information visualization*.