Clinical solid waste classification: an Artificial Intelligence-based approach

Hugo Gomes 58536, Marta Moreira 54459, Telma Veloso 58521

Abstract

In the current healthcare panorama, clinical solid waste management is a poorly regarded matter both in developed and underdeveloped countries, in the sense that some serious gaps are still in need of amendment. Due to the potencial hazards arising from the adoption of inadequate strategies, there is an urgent need to develop tools capable of improving the performance of the clinical solid waste classification process in particular. An artificial intelligence-driven approach for dealing with this issue is analysed and developed based on the portuguese clinical solid waste classification system.

Keywords: Artificial Intelligence, Clinical Solid Waste, Degree-of-Confidence, Extended Logic Programming, Knowledge Representation and Reasoning, Quality-of-Information, Waste Management

1. Introduction

As defined by the Medical Waste Tracking Act of 1988, clinical solid waste consist of solid waste materials which are generated during diagnosis, treatment, vaccination, research or in the production or testing of biological products for humans and animals, including syringes, live vaccines, blood and other waste contaminated with bodily fluids and removed body organs, among others. This class of waste is now recognized as a potencially hazardous agent affecting both the environment and the human being, due to the labor intensive operations entailed in its collection, segregation and disposal, which involve many possibilities of direct contact with the waste and therefore increase the risk of infections for healthcare workers, the general public and waste handlers in particular [1, 2]. Focusing on the latter, poor management practices and improper precautions taken by clinical waste workers during these operations are quoted as being the main reason of the spread of infectious diseases among clinical waste handlers [3, 4], which raises the additional need for adequate risk management strategies ensuring assessment, control, review and identification of risk. Nevertheless, several studies [5, 3, 6, 4] indicate that the clinical solid waste management at healthcare facilities is still inadequate in developed countries and that, in many situations, this class of waste is handled and disposed together with non-clinical waste. Given this scenario, the development of strategies for the definition of the best appropriate clinical waste management practice towards the minimization of occupational incidents and environmental contamination is of great importance. One of the main concerns still requiring addressing is the lack of awareness of both healthcare and clinical waste workers in what regards waste differential classification [1].

Taking into consideration the urgent need to bridge and amend these gaps, the present study aims at analysing and developing an artificial intelligence-driven approach for dealing with the judgement difficulties that arise from the waste classification process, especially in environments with defective information, based on the approach presented by *Neves et. al* in [7] and selectively focusing on the clinical waste management situation in Portugal.

2. Case Study

In Portugal, the management of clinical solid waste is regulated by law, and its legal constraints are defined within the Portuguese Legislative Decree no. 178/2006 - September 5th [8]. Particularly, this document provides a classification system for clinical solid waste based on four distinct criteria – typology, danger, production site and treatment required [9] –, which allow for a clear assortment of waste samples in four main classes (Figure 1), based on their specific combination of criteria specifications.

Class	Class Description
C1	Equivalent to Municipal Solid Waste (MSW).
C2	Non-biohazardous hospital waste.
C3	Biohazardous hospital waste.
C4	Specific hospital waste.

Figure 1: Portuguese classification system for clinical solid waste.

According to the classification criteria aforementioned, a database model was constructed (Figure 3), comprising a primary table and three secondary tables which refer to the analysed cases (Figure 2) and each of the considered criteria, respectively.

ID_Waste	Case Description
1	Scalpel contaminated with blood, provenient from the operating room.
2	Food waste collected from the maternity services.
3	Non-contaminated plaster piece from the orthopedic services.
4	Pair of medical gloves used in waste handling, contaminated with blood, organic fluids or both.
5	Package contaminated with cytostatic agents, either provenient from the labs or medical services.
6	Non-contaminated generic package, provenient from an unknown source.

Figure 2: Description of the cases analysed in the study.

ID_Waste	Class	ID_Source	ID_Type	ID_Contamination
1	C	6	1	1
2	C	4	2	5
3	C	6	4	5
4	C	3	6	{2, 3}
5	C	{5, 6}	3	4
6	C	⊥	1	5

ID_Source	GS	SS	WS	HS	LS	MS
1	1	0	0	0	0	0
2	0	1	0	0	0	0
3	0	0	1	0	0	0
4	0	0	0	1	0	0
5	0	0	0	0	1	0
6	0	0	0	0	0	1

ID_Type	GP	FW	CP	OM	DP	IP	MW	NB	SW	IB	MD	CH	AC	CO
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0	0	0	0	0	0	0
			...				...				...			
10	0	0	0	0	0	0	0	0	0	1	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	1	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	1	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	1

ID_Contamination	BL	OF	IA	CA
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	0	0	0	0
	...		...	
12	1	1	1	0
13	1	0	1	1
14	0	1	1	1
15	1	1	0	1
16	1	1	1	0

Figure 3: Database model (*vide* Appendix A for an extensive description of each acronym used.

2.1. Knowledge Representation and Reasoning

In regular Logic Programming (LP), the negative information is implicit – in other words, it is not possible to explicity state falsity and propositions are assumed false if there is no reason to believe otherwise. However, explicit negative information plays an important role in natural discourse and commonsense reasoning, and so, for use in deductive databases, knowledge representation and non-monotonic reasoning, a second kind of negation is included, giving rise to the Extended Logic Programming (ELP) paradigm [10]. An extended logic program is a finite set of clauses in the form:

$$\mathtt{q} \leftarrow \mathtt{q}_1 \wedge \mathtt{p}_n\ \mathtt{not}\ \mathtt{q}_1 \wedge \ldots \wedge \mathtt{not}\ \mathtt{q}_m$$
$$?\mathtt{p}_1 \wedge \ldots \wedge \mathtt{p}_n \wedge \mathtt{not}\ \mathtt{q}_1 \wedge \ldots \wedge \mathtt{not}\ \mathtt{p}_m\ (\mathtt{n,m} \geqslant 0)$$

where `?` is a domain denoting falsity, $\mathtt{p}_i$, $\mathtt{q}_j$ and `q` represent classical ground literals – either positive atoms or atoms preceeded by the classical negation sign [11]. In this representation formalism, every program is associated with a set of abducibles [12, 13], given here in the form of exceptions to the extensions of the predicates that compose the program.

In order to reason about the body of knowledge presented through the analysed cases, the relations defined in the database model were first rewritten in terms of the following predicates:

```
waste: ID_WS x Class x ID_SR x ID_TY x ID_CT

source: ID_SR x GS x SS x WS x HS x LS x MS

type: ID_TY x GP x FW x CP x OM x DP x OM x IP x MW x NB x SW x IB x MD x CH x AC x CO

contamination: ID_CT x BL x OF x IA x CA
```

Subsequently, the extension of the predicates was set in the form of four programs. Program 1 encloses the closure of the predicate `waste` and the declaration of each of the analysed cases, three of which refer to incomplete knowledge. With regard to the latter, different rules where created in order to address not only the unknown knowledge represented by the null value (⊥), but also the imprecise knowledge for both disjoint and not disjoint sets of abducibles, since in this particular universe of discourse an instance of `waste` can have multiple values for `contamination` but can only be associated with one value of `source`. Programs 2 through 4 are of less complex construction and define exclusively the closure of the predicates and the declaration of the possible combination of criteria specification, as the respective predicates do not contemplate any cases of incompleteness.

Program 1 Extended logic program for the predicate waste.

```
{
        ¬ waste (ID_WS, CL, ID_SR, ID_TY, ID_CT) ←
        not(waste (ID_WS, CL, ID_SR, ID_TY, ID_CT)),
        not(abducible_waste (ID_WS, CL, ID_SR, ID_TY, ID_CT)).
        waste(1,C,6,1,1).
        waste(2,C,4,2,5).
        waste(3,C,6,4,5).
        waste(4,C,3,6,{2,3}).
        waste(5,C,{5,6},3,4).
        waste(6,C,⊥,1,5).
abducible_waste (4, C, 3, 6, 2).
abducible_waste (4, C, 3, 6, 3).
abducible_waste (5, C, 5, 3, 4).
abducible_waste (5, C, 6, 3, 4).
?((abducible_waste (ID_WS, CL, ID_SR1, ID_TY, ID_CT)
∨
abducible_waste (ID_WS, CL, ID_SR2, ID_TY, ID_CT))
∧
¬ (abducible_waste (ID_WS, CL, ID_SR1, ID_TY, ID_CT)
∧
abducible_waste (ID_WS, CL, ID_SR2, ID_TY, ID_CT)))
abducible_waste (6, C, ⊥, 1, 5).
abducible_waste (ID_WS, CL, ID_SR, ID_TY, ID_CT) ← waste (ID_WS, CL, ⊥, ID_TY, ID_CT).
}
```

Program 2 Extended logic program for the predicate source.

```
{
        ¬ source (ID_SR, GS, SS, WS, HS, LS, MS) ←
        not(source (ID_SR, GS, SS, WS, HS, LS, MS)),
        not(abducible_source (ID_SR, GS, SS, WS, HS, LS, MS)).
        source(1,1,0,0,0,0,0).
        source(2,0,1,0,0,0,0).
        source(3,0,0,1,0,0,0).
        source(4,0,0,0,1,0,0).
        source(5,0,0,0,0,1,0).
        source(6,0,0,0,0,0,1).
}
```

Program 3 Extended logic program for the predicate type.

```
{
    ¬ type (ID_TY, GP,FW, CP, OM, DP, IP, MW, NB, SW, IB, MD, CH, AC, CO) ←
```

```
        not(type (ID_TY, GP,OM, DP, FW, IP, CP, MW, IB, NB, SW, MD, CH, AC, CO)),
    not(abducible_type (ID_TY, GP,OM, DP, FW, IP, CP, MW, IB, NB, SW, MD, CH, AC, CO)).
                     type(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0).
                     type(2,0,1,0,0,0,0,0,0,0,0,0,0,0,0).
                     type(3,0,0,1,0,0,0,0,0,0,0,0,0,0,0).
                     type(4,0,0,0,1,0,0,0,0,0,0,0,0,0,0).
                     type(5,0,0,0,0,1,0,0,0,0,0,0,0,0,0).
                                    ...
                     type(10,0,0,0,0,0,0,0,0,0,1,0,0,0,0).
                     type(11,0,0,0,0,0,0,0,0,0,0,1,0,0,0).
                     type(12,0,0,0,0,0,0,0,0,0,0,0,1,0,0).
                     type(13,0,0,0,0,0,0,0,0,0,0,0,0,1,0).
                     type(14,0,0,0,0,0,0,0,0,0,0,0,0,0,1).
}
```

Program 4 Extended logic program for the predicate `contamination`.

```
{
            ¬ contamination (ID_CT , BL, OF, IA, CA) ←
            not(contamination (ID_CT , BL, OF, IA, CA),
      not(abducible_contamination (ID_CT , BL, OF, IA, CA)).
                  contamination(1,1,0,0,0).
                  contamination(2,0,1,0,0).
                  contamination(3,0,0,1,0).
                  contamination(4,0,0,0,1).
                  contamination(5,0,0,0,0).
                             ...
                  contamination(12,1,1,1,0).
                  contamination(13,1,0,1,1).
                  contamination(14,0,1,1,1).
                  contamination(15,1,1,0,1).
                  contamination(16,1,1,1,1).
}
```

The following step consisted of defining the set of rules and actions that support the framework of the classification process, considering the specific combinations of criteria that characterize each of the four classes of waste.

Class 1

```
          [waste (ID, Class, SR, TY, CT), SR < 3, TY < 3, CT = 5]
                                    ↓
  [retract(waste(ID, Class, SR, TY, CT)), assert(waste(ID, C1, SR, TY, CT))].
```

Class 2

```
[waste (ID, Class, SR, TY, CT), SR = 6, 2 < TY < 7, CT = 5]
↓
[retract(waste(ID, Class, SR, TY, CT)), assert(waste(ID, C2, SR, TY, CT))].
```

Class 3

```
[waste (ID, Class, SR, TY, CT), SR > 2, 3 < TY < 10, CT < 4]
↓
[retract(waste(ID, Class, SR, TY, CT)), assert(waste(ID, C3, SR, TY, CT))].
```

Class 4

```
[waste (ID, Class, SR, TY, CT), SR > 4, TY > 9, CT = 4]
↓
[retract(waste(ID, Class, SR, TY, CT)), assert(waste(ID, C4, SR, TY, CT))].
```

Stopping Condition

```
[waste (ID, Class, SR, TY, CT)] → [print(Class),stop].
```

After running the analysed cases through the aforepresented set, the resulting knowledge resembles that represented in Figure 4.

ID_Waste	Class	ID_Source	ID_Type	ID_Contamination
1	C4	6	1	1
2	C1	4	2	5
3	C2	6	4	5
4	C3	3	6	{2, 3}
5	C4	{5, 6}	3	4
6	C1	$\perp$	1	5

Figure 4: Analysed cases after the classification process.

2.2. Quality-of-Information and Degree-of-Confidence

Since the present study aims at developing an approach which could ultimately offer user support in decision-making processes, it is important not only to address the choice of an adequate knowledge representation and reasoning paradigm but also to consider a means to assess the qualitative aspects of the information, especially when dealing with cases of incomplete knowledge. A measure of these qualitative aspects – the quality-of-information (Q_oI) – within logic programs has been object of some work with promising results [14, 15, 16]. With respect to the extension of a predicate i, the Q_oI [13] is given by a truth-value in the interval [0,1], depending on the extent to which the information is available: if it is known, regardless of being positive or negative, the truth value is 1; if it is unknown, the truth value is 0, according to the formula presented in (1), where `N` denotes the cardinality of the set of terms or clauses of the extension of predicate i that stand for the incompleteness considered.

$$QoI_i = \lim_{N \to \infty} \frac{1}{N} = 0 \tag{1}$$

For situations where the extension of predicate i is unknown but can be taken from a set of values, the Q_oI calculation method depends upon the characteristics of the abducible set under consideration. Thus, formulas (2) and (3) are alternatively used for disjoint and not disjoint abducible sets, respectively, where X denotes the cardinality of the abducible set for i whereas C_X^X stands for a card-combination subset with X elements.

$$QoI_i = \frac{1}{X} \tag{2}$$

$$QoI_i = \frac{1}{C_1^X + C_2^X + \ldots + C_X^X} \tag{3}$$

Taking this new variable into consideration, and including an additional variable which denotes one's confidence in a particular term of the extension of predicate i – the degree-of-confidence (D_oC) –, the closures of predicates previously declared in the extended logic programs were rewritten as follows:

Program 1 Rewritten closure of the predicate `waste`.

```
¬ waste(QoI, ID, CL, SR, TY, CT, DoC) ←
not(waste(QoI, ID, CL, SR, TY, CT, DoC)),
not(abducible_waste(QoI, ID, CL, SR, TY, CT, DoC)).
```

Program 2 Rewritten closure of the predicate `source`.

```
¬ source(QoI, ID, GS, SS, WS, HS, LS, MS, DoC) ←
```

```
not(source(QoI, ID, GS, SS, WS, HS, LS, MS, DoC)),
not(abducible_source(QoI, ID, GS, SS, WS, HS, LS, MS, DoC)).
```

Program 3 Rewritten closure of the predicate `type`.

```
¬ type(QoI, ID, GP, FW, CP, OM, DP, IP, MW, NB, SW, IB, MD, CH, AC, CO, DoC) ←
not(type(QoI, ID, GP, FW, CP, OM, DP, IP, MW, NB, SW, IB, MD, CH, AC, CO, DoC)),
not(abducible_type(QoI, ID, GP, FW, CP, OM, DP, IP, MW, NB, SW, IB, MD, CH, AC, CO,
DoC)).
```

Program 4 Rewritten closure of the predicate `contamination`.

```
¬ contamination(QoI, ID, BL, OF, IA, CA, DoC) ←
not(contamination(QoI, ID, BL, OF, IA, CA, DoC)),
not(abducible_contamination(QoI, ID, BL, OF, IA, CA, DoC)).
```

Considering that the values of all the criteria specifications are known, the values of both Q_oI and D_oC for every possible combination of the predicates `source`, `type` and `contamination` amount to 1. However, since the case study presented encloses situations of incomplete knowledge, the resulting six scenarios have, in some cases, a value for the mentioned parameters inferior to 1 or equal to 0, whenever the information is imprecise or unknown, respectively.

Scenario 1

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.33, 4, C3, 3, 6, 2, 0.87).
abducible_waste(0.5, 5, C4, 5, 3, 4, 0.9).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.8).
```

Scenario 2

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.33, 4, C3, 3, 6, 3, 0.87).
abducible_waste(0.5, 5, C4, 5, 3, 4, 0.9).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.8).
```

Scenario 3

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.17, 4, C3, 3, 6, 2, 0.834).
abducible_waste(0.17, 4, C3, 3, 6, 3, 0.834).
abducible_waste(0.5, 5, C4, 5, 3, 4, 0.9).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.8).
```

Scenario 4

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.33, 4, C3, 3, 6, 2, 0.834).
abducible_waste(0.5, 5, C4, 6, 3, 4, 0.9).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.8).
```

Scenario 5

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.33, 4, C3, 3, 6, 3, 0.834).
abducible_waste(0.5, 5, C4, 6, 3, 4, 0.9).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.8).
```

Scenario 6

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.17, 4, C3, 3, 6, 2, 0.834).
abducible_waste(0.17, 4, C3, 3, 6, 3, 0.834).
abducible_waste(0.5, 5, C4, 6, 3, 4, 0.9).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.8).
```

Based on the values of Q_oI obtained, two charts were constructed (Figure 5) in order to depict the qualitative characterization of each of the criteria specifications associated with incompleteness, namely the source of waste sample `ID_WS=5` and the contamination of waste sample `ID_WS=4`.

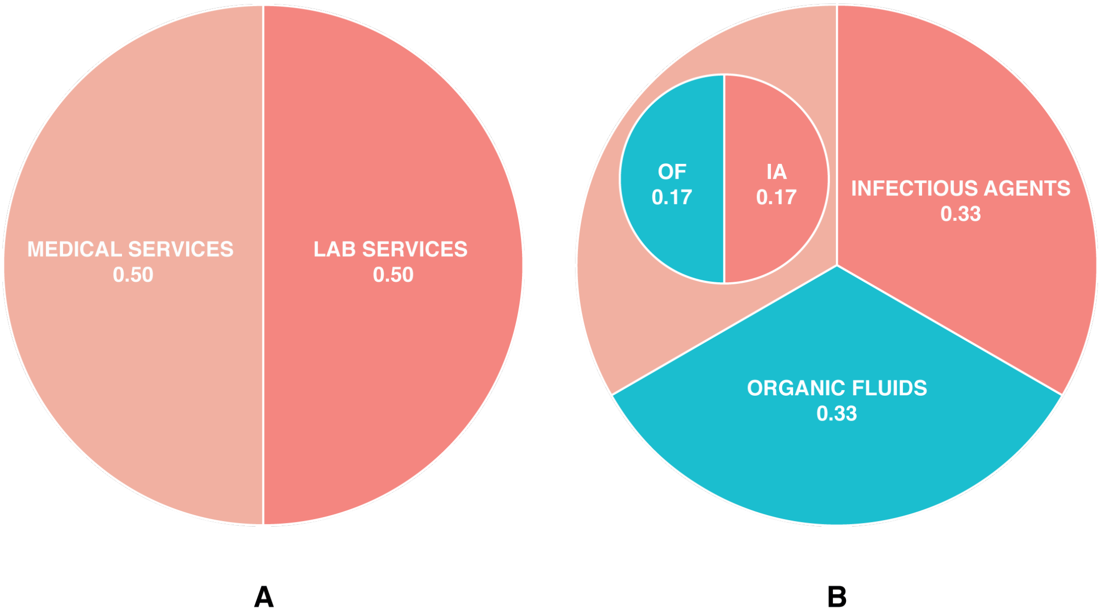

Figure 5: Charts depicting the quality-of-information for **A)** the source of waste sample `ID=5` and **B)** the contamination of waste sample `ID=4`.

Thus far, the value of the D_oC was computed considering that all of the criteria were equally weighted, according to the formula presented in (4).

$$DoC = \frac{\Sigma\ Q_oI(x_i)}{i\ terms} \tag{4}$$

$$DoC = \frac{\Sigma\ (w_i \times QoI(x_i))}{i\ terms} \tag{5}$$

However, in a more realistic scenario, each of the different criteria may influence the classification in a comparatively distinctive way, in which case the weights will not be equally distributed. Regardless of not having any proof that some criteria may overpower the remainder, in this particular case study, `contamination` was ascribed a weight of 0.4, in the alternative approach to the computation of the D_oC described in (5), due to the fact that, during the process of classification, classes `C1` and `C2` are instantly ruled out in case a waste sample is associated with some degree of contamination. As a result, the scenarios where rewritten using the newly computed D_oC as follows:

Scenario 1 Rewritten scenario for the normalized weighting approach.

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
```

$\texttt{abducible}_{waste}\texttt{(0.33, 4, C3, 3, 6, }\mathbf{2}\texttt{, 0.732).}$
$\texttt{abducible}_{waste}\texttt{(0.5, 5, C4, }\mathbf{5}\texttt{, 3, 4, 0.925).}$
$\texttt{abducible}_{waste}\texttt{(0, 6, C1, }\bot\texttt{, 1, 5, 0.85).}$

Scenario 2 Rewritten scenario for the normalized weighting approach.

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
```
$\texttt{abducible}_{waste}\texttt{(0.33, 4, C3, 3, 6, }\mathbf{3}\texttt{, 0.732).}$
$\texttt{abducible}_{waste}\texttt{(0.5, 5, C4, }\mathbf{5}\texttt{, 3, 4, 0.925).}$
$\texttt{abducible}_{waste}\texttt{(0, 6, C1, }\bot\texttt{, 1, 5, 0.85).}$

Scenario 3 Rewritten scenario for the normalized weighting approach.

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
```
$\texttt{abducible}_{waste}\texttt{(0.17, 4, C3, 3, 6, }\mathbf{2}\texttt{, 0.668).}$
$\texttt{abducible}_{waste}\texttt{(0.17, 4, C3, 3, 6, }\mathbf{3}\texttt{, 0.668).}$
$\texttt{abducible}_{waste}\texttt{(0.5, 5, C4, }\mathbf{5}\texttt{, 3, 4, 0.925).}$
$\texttt{abducible}_{waste}\texttt{(0, 6, C1, }\bot\texttt{, 1, 5, 0.85).}$

Scenario 4 Rewritten scenario for the normalized weighting approach.

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
```
$\texttt{abducible}_{waste}\texttt{(0.33, 4, C3, 3, 6, }\mathbf{2}\texttt{, 0.732).}$
$\texttt{abducible}_{waste}\texttt{(0.5, 5, C4, }\mathbf{6}\texttt{, 3, 4, 0.925).}$
$\texttt{abducible}_{waste}\texttt{(0, 6, C1, }\bot\texttt{, 1, 5, 0.85).}$

Scenario 5 Rewritten scenario for the normalized weighting approach.

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
```
$\texttt{abducible}_{waste}\texttt{(0.33, 4, C3, 3, 6, }\mathbf{3}\texttt{, 0.732).}$

```
abducible_waste(0.5, 5, C4, 6, 3, 4, 0.925).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.85).
```

Scenario 6 Rewritten scenario for the normalized weighting approach.

```
waste(1, 1, C4, 6, 1, 1, 1).
waste(1, 2, C1, 4, 2, 5, 1).
waste(1, 3, C2, 6, 4, 5, 1).
abducible_waste(0.17, 4, C3, 3, 6, 2, 0.668).
abducible_waste(0.17, 4, C3, 3, 6, 3, 0.668).
abducible_waste(0.5, 5, C4, 6, 3, 4, 0.925).
abducible_waste(0, 6, C1, ⊥, 1, 5, 0.85).
```

The differences between both computation approaches are illustrated in Figure 6, which reports solely to `Scenario 1` as a means of exemplification. There is a significant disparity in the values of D_oC obtained for the waste sample `ID_WS=4` – a pair of medical gloves used in waste handling associated with a set of not disjoint abducibles. When the contamination value is imprecise, as is the case, the D_oC value drops comparatively to the unweighted computation due to the fact that this criterion has the heaviest weight and a Q_oI value inferior to 1.

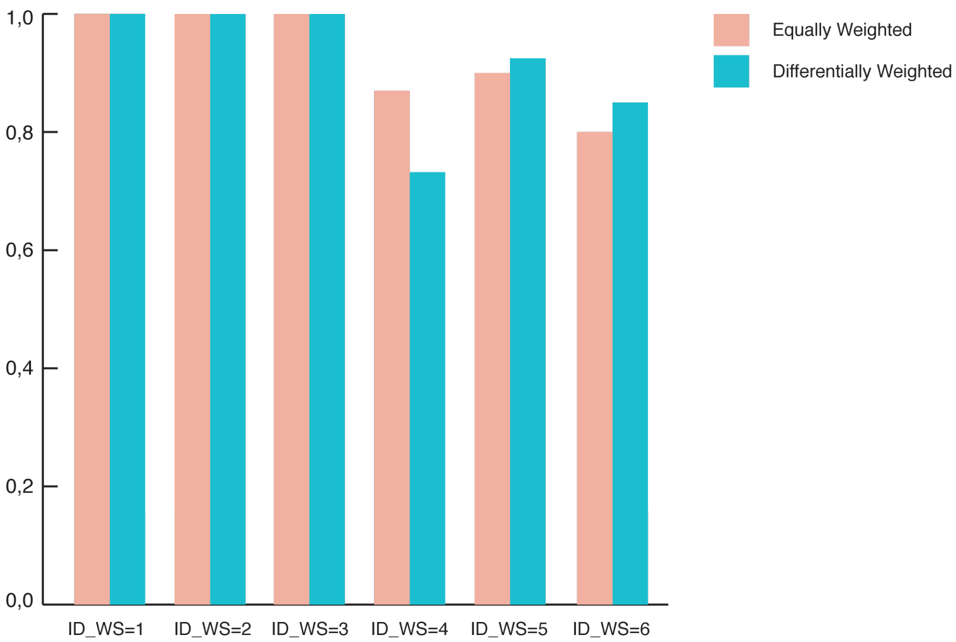

Figure 6: Chart depicting the D_oC value for both the situations when the criteria are equally weighted and when the `contamination` criterion is given a comparatively heavier weight of 0.4. The considered values refer to the two different versions obtained for `Scenario 1`.

3. Conclusions

Despite being of fundamental importance, clinical solid waste management is a poorly regarded matter within healthcare organizations, nowadays, in the sense that some serious gaps – mainly associated with a generalized lack of awareness – are still in need of amendment. Due to the potencial hazards arising from the adoption of inadequate strategies, there is an urgent need to develop tools capable of improving the performance of the clinical solid waste classification process and, as a consequence, reducing the human error associated with judgement difficulties in differential classification. In this study, an artificial intelligence-driven approach for dealing with this problem was analysed and developed based on the portuguese classification system. Concerning the fitting of the descriptive representation of the aforementioned system to a database model, some specificity had to be compromised in order to get a clear and objective representation.

In what regards knowledge representation and reasoning, the selected paradigm allows for an adequate treatment of cases enclosing incomplete knowledge and can easily be combined with both quality-of-information and degree-of-confidence quantification, which provide a means to assess qualitative aspects of the information under consideration. The computation of the degree-of-confidence was addressed in two distinctive ways, considering the initially formulated situation of equally weighted criteria and a more realistic situation in which the one of the criteria was given a comparatively heavier weight. The values obtained suggest that the normalized wheighting approach provides a more reliable depiction of the degree-of-confidence. Ultimately, this approach results in a set of possible classification scenarios strengthened with a measure of the confidence with which each one is associated, providing an hypothetic user with a solid decision making auxiliary support, and is intended to serve as a base for future work, namely the adaptation of the proposed model to a neural network representation modeling a decision support system, for instance. However, some additional regard would have to be considered concerning the criteria specifications, since there is a considerably large amount of plausible situations that are not envisioned in the current portuguese clinical solid waste classification system.

References

[1] S. Hossain, N. Rahman, V. Balakrishnan, Vi. Puvanesuaran, Z. Sarker, and M. Kadir. Infectious Risk Assessment of Unsafe Handling Practices and Management of Clinical Solid Waste. *International Journal of Environmental Research and Public Health*, 10(2):556–567, 2013.

[2] M. Akbolat, C. Dede, O. Isik, and H. Saglam. Medical Waste Management Practices in Turkey: A Case Study in Sakarya. *Pakistan Journal of Medical Sciences*, 27(4), 2011.

[3] I. Al-Khatib and C. Sato. Solid Health Care Waste Management Status at Health Care Centers in the West Bank–Palestinian Territory. *Waste Management*, 29(8):2398–403, 2009.

[4] A. Coker, A. Sangodoyin, M. Sridhar, C. Booth, P. Olomolaiye, and F. Hammond. Medical Waste Management in Ibadan, Nigeria: Obstacles and Prospects. *Waste Manag*, 29(2):804–11, 2009.

[5] M. El-Salam. Hospital Waste Management in El-Beheira Governorate, Egypt. *Journal of Environmental Management*, 91(3):618–629, 2010.

[6] F. Nemathaga, S. Maringa, and L. Chimuka. Hospital Solid Waste Management Practices in Limpopo Province, South Africa: a Case Study of Two Hospitals. *Waste Management*, 28(7):1236–1245, 2008.

[7] J. Neves, J. Ribeiro, P. Pereira, V. Alves, J. Machado, A. Abelha, P. Novais, C. Analide, M. Santos, and M. Fernández-Delgado. Evolutionary Intelligence in Asphalt Pavement Modeling and Quality-of-Information. *Progress in Artificial Intelligence*, 1(1):119–135, 2012.

[8] A. Botelho. Achieving compliance with healthcare waste management regulations: empirical evidence from small European healthcare units. 2011.

[9] Gestão de Resíduos Hospitalares para Unidades de Cuidados Continuados Integrados. Technical report, Ministério da Saúde, 2011.

[10] J. Alferes and L. Pereira. *Reasoning With Logic Programming*. Lecture Notes in Artificial Intelligence. Springer, 1111 edition, 1996.

[11] J. Neves. A Logic Interpreter to Handle Time and Negation in Logic Data Bases, booktitle = Proceedings of the 1984 Annual Conference of the ACM on The Fifth Generation Challenge, series = ACM '84, year = 1984, pages = 50–54, publisher = ACM.

[12] M. Denecker and A. C. Kakas. Abduction in Logic Programming. In *Computational Logic: Logic Programming and Beyond, Essays in Honour of Robert A. Kowalski, Part I*, pages 402–436. Springer-Verlag, 2002.

[13] J. Neves, J. Machado, C. Analide, A. Abelha, and L. Brito. The Halt Condition in Genetic Programming. In *Progress in Artificial Intelligence, 13th Portuguese Conference on Aritficial Intelligence, EPIA 2007, Workshops: GAIW, AIASTS, ALEA, AMITA, BAOSW, BI, CMBSB, IROBOT, MASTA, STCS, and TEMA, Guimares, Portugal, December 3-7, 2007, Proceedings*, volume 4874 of *Lecture Notes in Computer Science*, pages 160–169. Springer, 2007.

[14] P. Lucas. Quality Checking of Medical Guidelines Through Logical Abduction, booktitle = Procedures of AI-2003, Volume XX, year = 2003, pages = 309–321, publisher = Springer.

[15] A. Hommersom, P. Lucas, and P. Van Bommel. Checking the Quality of Clinical Guidelines Using Automated Reasoning Tools. *Theory and Practice of Logic Programming*, 8(5-6):611–641, November 2008.

[16] J. Machado, A. Abelha, P. Novais, J. Neves, and J. Neves. Quality of Service in Healthcare Units. *International Journal of Computer Aided Engineering and Technology*, 2(4):436–449, 2010.

Appendix A.

The complete description of each of the acronyms used to describe the terms involved in the specifications of the clinical solid waste classification criteria is given below (Figure A.7).

Acronym	
AC	Animal Cadavers
BL	Blood
CA	Cytostatic Agents
CH	Chemical (Products)
CO	Cutting (and Piercing) Objects
CP	Clinical Packages
DP	Diapers and Protections
FW	Food Waste
GP	General Packages
GS	General Services
HS	Hotel Services
IA	Infectious Agents
IB	Identified Body-Parts
IP	Individual Protection
LS	Lab Services
MD	Medical Drugs
MS	Medical Services
MW	Medical Waste
NB	Non-Identified Body-Parts
OF	Organic Fluids
OM	Ortopedic Material
SS	(Medical) Support Services
SW	(Medical) Support Waste
WS	Waste (Management) Services

Figure A.7: Acronyms used to describe each of the terms involved in the specifications of the clinical solid waste classification criteria.