

# Defeasible Reasoning with Prototype Descriptions: First Steps

Gabriele Sacco<sup>1,2</sup>, Loris Bozzato<sup>1</sup> and Oliver Kutz<sup>2</sup>

<sup>1</sup>Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy

<sup>2</sup>Free University of Bozen-Bolzano, Piazza Domenicani 3, 39100, Bolzano, Italy

## Abstract

The representation of defeasible information in Description Logics is a well-known issue and many formal approaches have been proposed, mostly emerging from existing formalisms in non-monotonic logic. However, in these proposals, little attention has been devoted to studying their capabilities in capturing the interpretation of typicality and exceptions from an ontological and cognitive point of view. In this regard, we are currently studying defeasible reasoning as discussed in the linguistic and cognitive literature in order to understand the important desiderata of defeasibility in commonsense reasoning.

In this paper, we provide an initial formalisation of a defeasible semantics for description logics which aims at fulfilling such desiderata. The proposal is based on combining ideas from prototype theory, weighted description logic (aka ‘tooth logic’), and earlier work on justifiable exceptions. The introduced weighted prototypes are normalised with respect to a given knowledge base, which in turn is used to compute a typicality score with respect to an individual. This machinery is then used to determine exceptions in case of conflicting axioms.

## Keywords


Description Logics, Weighted Logics, Perceptron Operators, Defeasible Reasoning

## 1. Introduction

Considering logic-based ontology representation languages, in Description Logics (DLs) many proposals for defining defeasibility and typicality have been formalised: as a matter of fact, most of them emerge from existing approaches in non-monotonic logics, as in [1, 2]. On the other hand, little attention has been devoted to study the capability of these approaches in capturing the interpretation of typicality and exceptions from the point of view of formal ontology and cognitive aspects. Consequently, the philosophical and cognitive assumptions of this kind of reasoning are often overlooked and need a committed discussion in order to understand the capabilities of the existing approaches.


Considering this, we recently initiated this discussion with an analysis of *generics* [3], sentences reporting a regularity regarding particular facts that can be generalised but tolerate exceptions. Our analysis (presented in [4]) highlighted three desiderata for non-monotonic reasoning:


---

 DL 2023: 36th International Workshop on Description Logics, September 2–4, 2023, Rhodes, Greece

 gsacco@fbk.eu (G. Sacco); bozzato@fbk.eu (L. Bozzato); Oliver.Kutz@unibz.it (O. Kutz)

 0000-0001-5613-5068 (G. Sacco); 0000-0003-1757-9859 (L. Bozzato); 0000-0003-1517-7354 (O. Kutz)

 © 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

- D1. *Exceptionality*: generics and non-monotonic reasoning both admit exceptions and much of the effort in the research has been dedicated to explain and model *how* exceptions can be tolerated. We think that another important aspect that should be considered is *why* something is an exception, i.e. how to include in the formal representation also the justification or explanation of why an instance is considered exceptional or not.
- D2. *Gradability*: normality is a graded notion in the case of typicality. For example, instead of typical individuals and atypical ones with respect to some concept, we have *more or less typical* individuals. For instance, it would not be possible to divide wolves between typical wolves and atypical ones in absolute terms, but there would be wolves that are more or less typical according to the specific features of each individual.
- D3. *Content sensitivity*: non-monotonic reasoning cannot be modelled by using only an extensional approach. This means that we cannot rely on purely extensional semantics, i.e. seeing the relation among concepts only in the light of relationships between sets. We need to take into account the semantics of the concepts involved in a broader sense, for example by relying on notions like typicality and saliency. The intuition here is that to explain why an individual is exceptional, for example, one would need some insights into the meaning (or, the content) of the statements of which the individual is an exception.

According to these desiderata, in this paper we sketch a new formal account for non-monotonic reasoning in DLs based on a graded reading of typicality, extending the work recently begun in [5]. Intuitively, in the case of a conflict between two facts about an individual, we can decide which one should be accepted according to *how much* the individual in question is typical w.r.t. such facts. For example: we know that dogs are trusted, whereas wolves are not; we know also that Balto is a wolfdog hybrid; we can ask ourselves now, should we infer that Balto is trusted or not? In our approach, we want to use the additional information we have about Balto being a dog and Balto being a wolf to see if he is a *more typical instance* of a dog or wolf and, according to this, infer if he is trusted or not.

More specifically, our approach is based on two main elements: *prototype definitions* and a *typicality score*. Prototype definitions are inspired by the prototype theory of concepts [6] and its representation based on the *tooth operator* as introduced, for example, in [7]. According to the endorsers of the *prototype theory* about concepts, being a member of a concept does not mean to satisfy a precise definition, but rather to satisfy enough features or constituents of that concept [8]. The second key element is the typicality score for individuals: this is calculated by inspecting to what extent the individual satisfies the *features* of the prototype. The aim of the score is to measure exactly how typical the individual is with respect to the prototype considered: in case of a conflict on prototype-related properties, the score provides a preference determining which conclusion should prevail for that specific individual.

From a technical point of view, our work also aims at investigating the use of DL weighted/-tooth operators in the context of defeasible reasoning, as hinted at in [9]. We remark that the current presentation of the formalization is still an initial proposal and includes some constraints to simplify its exposition: some of the possible refinements and extensions are briefly discussed in the conclusions.

## 2. DLs with Weighted Prototypes

On the basis of the ideas outlined above, we distinguish two parts in our knowledge bases: the actual DL knowledge base, which represents the knowledge of interest and can contain defeasible axioms and information about features of individuals, and a separate set containing prototype definitions.

In the following we outline the syntax and semantics of such enriched KBs.

### 2.1. Syntax: Features, Prototype Definitions, Prototype Knowledge Bases

The following definitions are independent from the DL language used for representing the main knowledge base: we consider a fixed concept language  $\mathcal{L}_\Sigma$  (such as for example  $\mathcal{ALC}$ ) based on a DL signature  $\Sigma$  with disjoint and non-empty sets  $\text{NC}$  of concept names,  $\text{NR}$  of role names, and  $\text{NI}$  of individual names. Furthermore, we identify a subset of the concept names as denoting *prototype names* by assuming a subset  $\text{NP} \subseteq \text{NC}$  and a set of *feature names*  $\text{NF} \subseteq \text{NC}$  with  $\text{NP} \cap \text{NF} = \emptyset$ .

**Definition 1** (Features). *A basic feature is a concept name  $C \in \text{NF}$ . A general feature is a complex concept in language  $\mathcal{L}_\Sigma$  using only basic features as concept names.*

For simplicity, we call *general concepts* the concepts composed only of concepts in  $\text{NC} \setminus \text{NP} \cup \text{NF}$ .

The features associated with prototypes together with the degree of their importance are given in *prototype definitions*.<sup>1</sup>

**Definition 2** (Positive prototype definition). *Let  $P \in \text{NP}$  be a prototype name, let  $C_1, \dots, C_m$  be general features of  $\mathcal{L}_\Sigma$  and let  $\bar{w} = (w_1, \dots, w_m) \in \mathbb{Q}^m$  be a weight vector of rational numbers, where for every  $i \in \{1, \dots, m\}$  we have  $w_i > 0$ . Then, the expression*

$$P(C_1 : w_1, \dots, C_m : w_m)$$

*is called a (positive) prototype definition for  $P$ .*

Intuitively, the weights associated to the features can be then combined to compute a score denoting the degree of typicality of an instance w.r.t. the prototype: for the current definition, weights are assumed to be positive and features are independent. We use here rational weights, which is sufficient for practical purposes. Real numbers could be allowed as well, but this would not substantially change the formal setup; this is also the case for the related perceptron logic [10].

Note that, since some features could be mutually exclusive (e.g. the color of an apple can be red or green, but not both), prototype definitions should not be seen as denoting a “perfect individual”. To allow for a direct comparison across scores of different prototypes, these need to be normalised to a common value interval, possibly with a scoring function that does not depend on the number of features defining different prototypes.

In an initial proposal, we simply constrained the weights of features to be in the  $[0, 1]$  interval and to prescribe further that they would add up to 1, i.e. prototypes were simply assumed to

---

<sup>1</sup>Note that this definition of prototypes is similar to the definition of concepts by the tooth operator defined in [7].

be given as *positive* and *normalised* [5]: in the following sections, we provide instead a more general proposal for normalising prototype scores.

In the knowledge part of the KB, we can use prototype names in DL axioms to describe properties of the members of such classes. Here we consider the case in which prototype names are only used as primitive concepts on the left hand side of concept inclusions.

In particular, we call a concept inclusion of the type  $P \sqsubseteq D$  a *prototype axiom* if  $P \in \text{NP}$  and  $D$  is a general concept of  $\mathcal{L}_\Sigma$ . Intuitively, these axioms are not absolute and can be “overridden” by prototype instances (cf. defeasible axioms in [11]), also depending on the “degree of membership” of the individual to the given prototype (i.e., the satisfaction of its features). Prototype axioms can be seen as corresponding to generic sentences since they express generalisations that admit exceptions. Such exceptions can thus override the truth of a prototype axiom for that specific individual.

As noted above, we consider knowledge bases which can contain prototype axioms and which are enriched with an accessory KB, the PBox  $\mathcal{P}$  providing prototype definitions.

**Definition 3** (Prototyped Knowledge Base, PKB). *A prototyped knowledge base, PKB for short, in language  $\mathcal{L}_\Sigma$  is a triple  $\mathfrak{K} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$  where:*

- $\mathcal{T} = T_P \uplus T_C \uplus T_F$  is a DL TBox consisting of concept inclusion axioms of the form  $C \sqsubseteq D$  and partitioned into the disjoint sets  $T_P$  of prototype axioms,  $T_C$  of general concept inclusions based on arbitrary concepts and  $T_F$  of feature axiom, strict subsumptions regarding features;
- $\mathcal{A} = A_P \uplus A_C \uplus A_F$  is a set of ABox assertions of the form  $C(a)$ , where  $a \in \text{NI}$  is an individual name, and partitioned into the disjoint sets  $A_P$  of prototype assertions (where  $C \in \text{NP}$ ),  $A_C$  of general assertions (where  $C$  is a general concept) and  $A_F$  of basic feature assertions (where  $C \in \text{NF}$ );
- $\mathcal{P}$  is a set of prototype definitions, exactly one for each prototype name  $P \in \text{NP}$  appearing in the prototype TBox  $T_P$ .

Note that a PKB  $\langle \mathcal{T}, \mathcal{A}, \emptyset \rangle$  can be seen as a standard DL knowledge base.

**Example 1.** *We can now represent the example described in the introduction as a prototyped knowledge base  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$  as follows:*

$$\mathcal{T} = \{ \text{Dog} \sqsubseteq \text{Trusted}, \text{Wolf} \sqsubseteq \neg \text{Trusted}, \text{Dog} \sqsubseteq \text{hasLegs}, \text{Wolf} \sqsubseteq \text{hasLegs} \},$$

$$\begin{aligned} \mathcal{A} = \{ & \text{Dog}(\text{balto}), \text{Wolf}(\text{balto}), \text{Dog}(\text{pluto}), \text{Wolf}(\text{alberto}), \text{Dog}(\text{cerberus}), \\ & \text{livesInWoods}(\text{balto}), \text{hasLegs}(\text{balto}), \text{isTamed}(\text{balto}), \\ & \text{hasCollar}(\text{pluto}), \text{hasLegs}(\text{pluto}), \text{isTamed}(\text{pluto}), \\ & \text{hasLegs}(\text{alberto}), \text{Hunts}(\text{alberto}) \\ & \neg \text{Trusted}(\text{cerberus}) \}, \end{aligned}$$

$$\begin{aligned} \mathcal{P} = \{ & \text{Wolf}(\text{livesInWoods} : 10, \text{hasLegs} : 4, \text{livesInPack} : 8, \text{Hunts} : 11), \\ & \text{Dog}(\text{hasCollar} : 33, \text{livesInHouse} : 22, \text{hasLegs} : 11, \text{isTamed} : 44) \} \end{aligned}$$

Below we will construct a semantics for this kind of PKB which will entail and justify the conclusion that `balto` is a trusted dog which is a wolf, without being inconsistent, and that `cerberus` is an exceptional dog with respect to the property of dogs of being trusted. Note that in the case of the instances `pluto` and `alberto` no contradiction arises, thus we want that the axioms in  $\mathcal{T}$  are applied to them normally.  $\diamond$

## 2.2. Semantics for Prototype Knowledge Bases

The semantics of PKBs is based on standard interpretations for the underlying DL  $\mathcal{L}_\Sigma$ . However, we need to introduce additional semantic structure to manage exceptions to prototype axioms, exploiting the prototype definition expressions in  $\mathcal{P}$ .

**Definition 4** (PKB interpretations). A PKB interpretation is a description logic interpretation  $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$  for signature  $\Sigma$  with a non-empty domain,  $\Delta^{\mathcal{I}}, a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$  for every  $a \in \text{NI}$ ,  $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$  for every  $A \in \text{NC}$ ,  $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$  for every  $R \in \text{NR}$ , and where the extension of complex concepts is defined recursively as usual for language  $\mathcal{L}_\Sigma$ .

Note that we are not giving a DL interpretation to the prototype definition expressions in  $\mathcal{P}$ .

We consider the notion of axiom instantiation and clashing assumptions as defined in [11]: intuitively, for an axiom  $\alpha \in \mathcal{L}_\Sigma$  the instantiation of  $\alpha$  with  $e \in \text{NI}$ , written  $\alpha(e)$ , is the specialization of  $\alpha$  to  $e$ .

**Definition 5** (Clashing assumptions and clashing sets). A clashing assumption is a pair  $\langle \alpha, e \rangle$  such that  $\alpha(e)$  is an axiom instantiation of  $\alpha$ , and  $\alpha \in T_P$  is a prototype axiom.

A clashing set for  $\langle \alpha, e \rangle$  is a satisfiable set  $S$  of ABox assertions s.t.  $S \cup \{\alpha(e)\}$  is unsatisfiable.

Intuitively, a clashing assumption  $\langle P \sqsubseteq D, e \rangle$  states that we assume that  $e$  is an exception to the prototype axiom  $P \sqsubseteq D$  in a given PKB interpretation. Then, the fact that a clashing set  $S$  for  $\langle P \sqsubseteq D, e \rangle$  is verified by such an interpretation gives a “justification” of the validity of the assumption of overriding. This intuition is reflected in the definition of models: we first extend PKB interpretations with a set of clashing assumptions.

**Definition 6** (CAS-interpretation). A CAS-interpretation is a structure  $\mathcal{I}_{CAS} = \langle \mathcal{I}, \chi \rangle$  where  $\mathcal{I}$  is a PKB interpretation and  $\chi$  is a set of clashing assumptions.

Then, CAS-models for a PKB  $\mathfrak{K}$  are CAS-interpretations that verify “strict” axioms in  $T_C$  and  $T_F$  and defeasibly apply prototype axioms in  $T_P$  (excluding the exceptional instances in  $\chi$ ).

**Definition 7** (CAS-model). Given a PKB  $\mathfrak{K}$ , a CAS-interpretation  $\mathcal{I}_{CAS} = \langle \mathcal{I}, \chi \rangle$  is a CAS-model for  $\mathfrak{K}$  (denoted  $\mathcal{I}_{CAS} \models \mathfrak{K}$ ), if the following holds:

- (i) for every  $\alpha \in T_C \cup T_F \cup \mathcal{A}$  of  $\mathcal{L}_\Sigma$ ,  $\mathcal{I} \models \alpha$ ;
- (ii) for every  $\alpha = P \sqsubseteq D \in T_P$ , if  $\langle \alpha, d \rangle \notin \chi$ , then  $\mathcal{I} \models \alpha(d)$ .

Two DL interpretations  $\mathcal{I}_1$  and  $\mathcal{I}_2$  are NI-congruent, if  $c^{\mathcal{I}_1} = c^{\mathcal{I}_2}$  holds for every  $c \in \text{NI}$ . This extends to CAS interpretations  $\mathcal{I}_{CAS} = \langle \mathcal{I}, \chi \rangle$  by considering PKB interpretations  $\mathcal{I}$ . Intuitively, we say that a CAS-interpretation is justified if all of its clashing assumptions admit a clashing set that is verified by the interpretation.

**Definition 8** (Justifications). We say that  $\langle \alpha, e \rangle \in \chi$  is justified for a CAS-model  $\mathcal{I}_{CAS}$ , if some clashing set  $S_{\langle \alpha, e \rangle}$  exists such that, for every  $\mathcal{I}'_{CAS} = \langle \mathcal{I}', \chi \rangle$  of  $\mathfrak{K}$  that is NI-congruent with  $\mathcal{I}_{CAS}$ , it holds that  $\mathcal{I}' \models S_{\langle \alpha, e \rangle}$ . A CAS model  $\mathcal{I}_{CAS}$  of a PKB  $\mathfrak{K}$  is justified, if every  $\langle \alpha, e \rangle \in \chi$  is justified in  $\mathfrak{K}$ .

We define the consequence from justified CAS-models:  $\mathfrak{K} \models_{J_{CAS}} \alpha$  if  $\mathcal{I}_{CAS} \models \alpha$  for every justified CAS-model  $\mathcal{I}_{CAS}$  of  $\mathfrak{K}$ .

The main intuition of prototype definitions is that each instance of a prototype is associated with a score which denotes the “degree of typicality” of the individual with respect to the concept described by the prototype. As in [7], such a degree is computed from the prototype features that are satisfied by the instances and their score. Ideally, the prototype score of an individual allows us to determine a preference over models: axioms on prototypes with higher score are preferred to the ones on lower scoring prototypes; thus the measure needs to be independent of single models and comparable across different prototypes.

Formally, a simple score function can be defined as follows:

**Definition 9** (Prototype score). Given a prototype definition  $P(C_1 : w_1, \dots, C_m : w_m)$ , we define the score function  $score_P : \text{NI} \rightarrow \mathbb{R}$  for prototype  $P$  as:

$$score_P(a) = \sum_{\mathfrak{K} \models_{J_{CAS}} C_i(a)} w_i$$

This measure, however, depends on the value interval over which the prototype weights have been defined: in order to compare the score of an individual with scores relative to other prototypes, this value needs to be normalized. We do so by computing the maximum score  $max_P$  and minimum score  $min_P$  for all prototypes. The idea for this scoring function is derived from the so-called tooth-max operator introduced in [10] and applied for instance in [12] to some cognitive modelling problems such as over-extension and dominance of features. Here, the tooth-max is a concept description that collects all those individuals in a given model that obtain the maximal possible sum of feature weights, that is, that realise some specific value  $t$  which corresponds to the maximal realisable weight in this situation (different selections of feature combinations might result in this value  $t$ ). It was shown in [10] that this concept forming operation, when taken as a logical operator, is in fact equivalent to the universal modality, which is known to significantly extend expressivity of standard DLs or modal logics [13].

Coming back to the specifics of how we here want to compute the scoring function, the maximum score  $max_P$  denotes the score of the maximum value of  $score_P$  obtainable from the weights of consistent subset of features of  $P$ .<sup>2</sup> Formally, given a prototype definition  $P(C_1 : w_1, \dots, C_m : w_m)$ , let  $\mathcal{S}_P$  be the set of sets  $S_P \subseteq \{C_1, \dots, C_n\}$  s.t.  $S_P \cup \mathfrak{K}$  is consistent. Then:

$$max_P = \max\left(\sum_{C_i \in S_P} w_i \mid S_P \in \mathcal{S}_P\right)$$

The minimal score  $min_P$  denotes the sum of the weights for “unavoidable” features, namely

<sup>2</sup>We note that the computation of maximum score is related to the idea of maximization operator in [10].

those that are strictly implied by the membership to the prototype concept. Formally:

$$\min_P = \sum_{\mathfrak{R} \models_{JCAS} P \sqsubseteq C_i} w_i$$

A normalized score function  $nscore_P$  can be derived from  $score_P$  as:

$$nscore_P(a) = \frac{score_P(a) - \min_P}{\max_P}$$

The normalized scoring function can then be used to define preferences over models: in particular, we want to prefer justified CAS models where the *exceptions* appear on elements of the *less* scoring prototypes. This can be encoded as follows:

**Definition 10** (Preference SP).  $\chi_1 > \chi_2$  if, for every  $\langle P \sqsubseteq D, e \rangle \in \chi_1 \setminus \chi_2$  such that there exists a  $\langle Q \sqsubseteq E, e \rangle \in \chi_2 \setminus \chi_1$  with  $\mathfrak{R} \cup \{D(e), E(e)\}$  unsatisfiable, it holds that  $nscore_{\mathfrak{R}}^P(e) < nscore_{\mathfrak{R}}^Q(e)$ .

The intuition behind the condition  $\mathfrak{R} \cup \{D(e), E(e)\}$  unsatisfiable is that we want to make the comparison between the clashing assumptions that are directly in conflict.

Given two CAS-interpretations  $\mathcal{I}_{CAS}^1 = \langle \mathcal{I}^1, \chi_1 \rangle$  and  $\mathcal{I}_{CAS}^2 = \langle \mathcal{I}^2, \chi_2 \rangle$ , we say that  $\mathcal{I}_{CAS}^1$  is preferred to  $\mathcal{I}_{CAS}^2$  (denoted  $\mathcal{I}_{CAS}^1 > \mathcal{I}_{CAS}^2$ ) if  $\chi_1 > \chi_2$ .

Finally, we define the notion of PKB model as a minimal justified model for the PKB.

**Definition 11** (PKB model). An interpretation  $\mathcal{I}$  is a PKB model of  $\mathfrak{R}$  (denoted,  $\mathcal{I} \models \mathfrak{R}$ ) if

- $\mathfrak{R}$  has some justified CAS model  $\mathcal{I}_{CAS} = \langle \mathcal{I}, \chi \rangle$ .
- there exists no justified  $\mathcal{I}'_{CAS} = \langle \mathcal{I}', \chi' \rangle$  that is preferred to  $\mathcal{I}_{CAS}$ .

The consequence from PKB models of  $\mathfrak{R}$  (denoted  $\mathfrak{R} \models \alpha$ ) allows us to use the degree of typicality of instances to verify which of the conflicting prototype axioms should apply.

**Example 2.** Considering the PKB reported in the example above, assume to have two PKB interpretations  $\mathcal{I}_1$  and  $\mathcal{I}_2$  associated respectively with the following two sets of clashing assumptions

$$\begin{aligned} \chi_1 &= \{ \langle \text{Wolf} \sqsubseteq \neg \text{Trusted}, \text{balto} \rangle, \langle \text{Dog} \sqsubseteq \text{Trusted}, \text{cerberus} \rangle \} \text{ and} \\ \chi_2 &= \{ \langle \text{Dog} \sqsubseteq \text{Trusted}, \text{balto} \rangle, \langle \text{Dog} \sqsubseteq \text{Trusted}, \text{cerberus} \rangle \}. \end{aligned}$$

We have now two CAS-interpretations corresponding to  $\langle \mathcal{I}_1, \chi_1 \rangle$  and  $\langle \mathcal{I}_2, \chi_2 \rangle$ . Assuming that they are also CAS-models, we can check if the two are also justified. Since, the clashing assumptions have the following clashing sets, respectively  $\{ \text{Wolf}(\text{balto}), \text{Trusted}(\text{balto}), \text{Dog}(\text{cerberus}), \neg \text{Trusted}(\text{cerberus}) \}$  for the clashing assumptions in  $\chi_1$  and  $\{ \text{Dog}(\text{balto}), \neg \text{Trusted}(\text{balto}), \text{Dog}(\text{cerberus}), \neg \text{Trusted}(\text{cerberus}) \}$  for those in  $\chi_2$ , they are both justified.

In order to decide which model is preferred, we need to compute the prototype scores for *balto* and for *cerberus*: we have  $score_{\text{Wolf}}(\text{balto}) = 14$ ,  $score_{\text{Dog}}(\text{balto}) = 55$ ,  $score_{\text{Dog}}(\text{cerberus}) = 11$ . Then we need to normalise them:

$$nscore_{\text{Dog}}(\text{balto}) = \frac{score_{\text{Dog}}(\text{balto}) - \min_{\text{Dog}}}{\max_{\text{Dog}}} = \frac{55 - 11}{110} = 0,4$$

$$nscore_{Wolf}(balto) = \frac{score_{Wolf}(balto) - min_{Wolf}}{max_{Wolf}} = \frac{14 - 4}{33} \approx 0,3$$

$$nscore_{Dog}(cerberus) = \frac{score_{Dog}(cerberus) - min_{Dog}}{max_{Dog}} = \frac{11 - 11}{33} = 0$$

Consequently  $score_{Wolf}(balto) < score_{Dog}(balto)$  and, since  $\langle Dog \sqsubseteq Trusted, cerberus \rangle$  is present in both  $\chi_1$  and  $\chi_2$  so it does not influence the preference order, then we can conclude that  $\chi_2 > \chi_1$ . This means that the preferred model, i.e. the only PKB model, is  $\mathcal{I}_1$  where  $balto$  is an exception to  $Wolf \sqsubseteq \neg Trusted$  and  $cerberus$  is an exception to  $Dog \sqsubseteq Trusted$ . Consequently, it holds that  $\mathfrak{R} \models Trusted(balto)$  and  $\mathfrak{R} \models \neg Trusted(cerberus)$ .

Moreover, we can note that for  $pluto$  and  $alberto$  we can standardly infer  $Trusted(pluto)$  and  $\neg Trusted(alberto)$ . The reason is that the clashing assumptions are referred to specific individuals, and since there are no contradicting assertions for  $pluto$  and  $alberto$ , there are no clashing sets that justify their assumptions as exceptions. Therefore, axioms in  $\mathcal{T}$  apply to them standardly.  $\diamond$

We note that, like PKB models can be related to Answer Sets (different solutions under different assumptions for exceptions), the kind of ordering on the models is akin to the Answer Set preferences definable with weak constraints or Asprin preferences. In fact, this has been used in the implementation of CKR with justified exceptions in [14, 15].

### 3. Related Works

As we said in the introduction, many formalisms for defeasible reasoning have been already developed in the framework of DLs. An extensive comparison with such approaches is currently out of the scope of this initial paper, but we can already draw some relations. Firstly, our work can be compared to more “classical” approaches like [16, 17]: these approaches are inspired by the historical work on defeasible reasoning in propositional logic presented in [18, 19], where formal properties, known as KLM properties, have been introduced as properties that any non-monotonic logic should satisfy.

Of particular interest for our work are formalisms developed starting from [16], which use weights and have a multi-preferential relation over the individuals with respect to the concepts they are instances of, as, for instance, [20, 21]. The interest comes from the fact that there are commonalities with our formalism since both exploit weights and introduce preference relations on the domain which are not absolute.

Other than works strictly about defeasibility in DLs, our approach can be compared also to works that share our interest for the results coming from cognitive science and philosophy to develop formal systems in the field of knowledge representation and in particular using the language of DLs. Examples of these works, particularly interested in the notion of typicality, are [22, 23].



## 4. Discussion and Conclusions

We presented an initial formalisation for a non-monotonic extension of DLs with the goal of satisfying three desiderata extracted from a critical discussion on generics and the prototype theory about concepts. We note that our formalism meets the desiderata: (*D1*). the formalisation is based on the idea that we need to justify an exception to an axiom by looking at how (a)typical an individual is: in other words, we use typicality to decide to which of the conflicting axioms (which correspond to generics) the individual is an exception; (*D2*). we are using a graded notion of typicality: we do not simply have typical and atypical individuals, but we compute a score which is comparable across prototypes; (*D3*). the notion of typicality we introduce is not extensional: by using the scores to represent it, and which contextually depend on the information in the knowledge base, we are relying on a characteristic that goes beyond a static and extensional set-theoretic treatment.

In future work, we want to extend the cognitive and ontological study of exceptions sketched here also by comparing it with other accounts for typicality and defeasibility in DLs in greater detail. Regarding our proposed formalisation, we need to further explore and refine the formal consequences and properties of our approach. In particular, we need to discuss what the best options are to compute the scores in order to have a balanced score for every prototype and how to extend this computation to roles, possibly following the line of work on counting perceptron logic presented in [24, 25]. Here, not only the satisfaction of a certain feature may give rise to a ‘weight contribution’ in the prototype, but each instance of a role filler might be individually contributing to the overall weight. Moreover, different readings of the weights could also give rise to alternative score functions, and particularly, the weights need not be added up in a linear additive way. Another extension of the formalism could involve the extension of the degree of typicality from prototypes to single axioms. Finally, we need to better understand how to allow for more interaction between concepts used for prototypes and features, for example by allowing nested definitions of prototypes, use prototype concepts as features and compute scores with defeasible features.

## References

- [1] L. Giordano, V. Gliozzi, A. Lieto, N. Olivetti, G. L. Pozzato, Reasoning about typicality and probabilities in preferential description logics, 2020. URL: <https://arxiv.org/abs/2004.09507>. doi:10.48550/ARXIV.2004.09507.
- [2] K. Britz, J. Heidema, T. Meyer, Modelling object typicality in description logics, in: A. Nicholson, X. Li (Eds.), *AI 2009: Advances in Artificial Intelligence*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 506–516.
- [3] S.-J. Leslie, Generics: Cognition and acquisition, *Philosophical Review* 117 (2008) 1–47.
- [4] G. Sacco, L. Bozzato, O. Kutz, Generics in defeasible reasoning, exceptionality, gradability, and content sensitivity, 2023. Accepted at 7th CAOS Workshop ‘Cognition and Ontologies’, 9th Joint Ontology Workshops (JOWO 2023), co-located with FOIS 2023, 19-20 July, 2023, Sherbrooke, Québec, Canada.
- [5] G. Sacco, L. Bozzato, O. Kutz, Introducing weighted prototypes in description logics for

- defeasible reasoning, in: A. F. Agostino Dovier (Ed.), Proceedings of the 38th Italian Conference on Computational Logic, CEUR Workshop Proceedings, Udine, Italy, 2023.
- [6] J. A. Hampton, Concepts as prototypes, volume 46 of *Psychology of Learning and Motivation*, Academic Press, 2006, pp. 79–113.
- [7] P. Galliani, G. Righetti, O. Kutz, D. Porello, N. Troquard, Perceptron connectives in knowledge representation, in: C. M. Keet, M. Dumontier (Eds.), Knowledge Engineering and Knowledge Management, Springer International Publishing, Cham, 2020, pp. 183–193.
- [8] E. Margolis, S. Laurence, Concepts, in: E. N. Zalta, U. Nodelman (Eds.), The Stanford Encyclopedia of Philosophy, Fall 2022 ed., Metaphysics Research Lab, Stanford University, 2022.
- [9] P. Galliani, O. Kutz, D. Porello, G. Righetti, N. Troquard, On knowledge dependence in weighted description logic, in: D. Calvanese, L. Iocchi (Eds.), GCAI 2019. Proceedings of the 5th Global Conference on Artificial Intelligence, volume 65 of *EPiC Series in Computing*, EasyChair, 2019, pp. 68–80.
- [10] D. Porello, O. Kutz, G. Righetti, N. Troquard, P. Galliani, C. Masolo, A toothful of concepts: Towards a theory of weighted concept combination, in: Description Logics, volume 2373 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019.
- [11] L. Bozzato, T. Eiter, L. Serafini, Enhancing context knowledge repositories with justifiable exceptions, *Artif. Intell.* 257 (2018) 72–126.
- [12] G. Righetti, P. Galliani, O. Kutz, D. Porello, C. Masolo, N. Troquard, Weighted Description Logic for Classification Problems, in: D. Calvanese, L. Iocchi (Eds.), GCAI 2019. Proceedings of the 5th Global Conference on Artificial Intelligence, volume 65 of *EPiC Series in Computing*, EasyChair, 2019, pp. 108–112. URL: <https://easychair.org/publications/paper/S5bV>. doi:10.29007/vd1q.
- [13] V. Goranko, S. Passy, Using the universal modality: Gains and questions, *Journal of Logic and Computation* 2 (1992) 5–30.
- [14] L. Bozzato, T. Eiter, R. Kiesel, Reasoning on multirelational contextual hierarchies via answer set programming with algebraic measures, *Theory Pract. Log. Program.* 21 (2021) 593–609. URL: <https://doi.org/10.1017/S1471068421000284>. doi:10.1017/S1471068421000284.
- [15] L. Bozzato, L. Serafini, T. Eiter, Reasoning with justifiable exceptions in contextual hierarchies, in: M. Thielscher, F. Toni, F. Wolter (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference, KR 2018, Tempe, Arizona, 30 October - 2 November 2018, AAAI Press, 2018, pp. 329–338. URL: <https://aaai.org/ocs/index.php/KR/KR18/paper/view/18032>.
- [16] L. Giordano, V. Gliozzi, N. Olivetti, G. Pozzato, Semantic characterization of rational closure: From propositional logic to description logics, *Artificial Intelligence* 226 (2015) 1–33. URL: <https://www.sciencedirect.com/science/article/pii/S0004370215000673>. doi:<https://doi.org/10.1016/j.artint.2015.05.001>.
- [17] K. Britz, G. Casini, T. Meyer, K. Moodley, U. Sattler, I. Varzinczak, Principles of klm-style defeasible description logics, *ACM Trans. Comput. Logic* 22 (2020). URL: <https://doi.org/10.1145/3420258>. doi:10.1145/3420258.
- [18] S. Kraus, D. Lehmann, M. Magidor, Nonmonotonic reasoning, preferential models and cumulative logics, *Artificial Intelligence* 44 (1990) 167–207. URL: [https://doi.org/10.1016/0001-8889\(90\)90010-9](https://doi.org/10.1016/0001-8889(90)90010-9).

www.sciencedirect.com/science/article/pii/S0004370290901015. doi:[https://doi.org/10.1016/0004-3702\(90\)90101-5](https://doi.org/10.1016/0004-3702(90)90101-5).

- [19] D. Lehmann, M. Magidor, What does a conditional knowledge base entail?, *Artificial Intelligence* 55 (1992) 1–60. URL: <https://www.sciencedirect.com/science/article/pii/S000437029290041U>. doi:[https://doi.org/10.1016/0004-3702\(92\)90041-U](https://doi.org/10.1016/0004-3702(92)90041-U).
- [20] L. Giordano, D. Theseider Dupré, Weighted defeasible knowledge bases and a multipreference semantics for a deep neural network model, in: *Logics in Artificial Intelligence: 17th European Conference, JELIA 2021, Virtual Event, May 17–20, 2021, Proceedings 17*, Springer, 2021, pp. 225–242.
- [21] L. Giordano, D. Theseider Dupré, An ASP approach for reasoning on neural networks under a finitely many-valued semantics for weighted conditional knowledge bases, *Theory and Practice of Logic Programming* 22 (2022) 589–605. doi:[10.1017/S1471068422000163](https://doi.org/10.1017/S1471068422000163).
- [22] A. Lieto, G. L. Pozzato, et al., What cognitive research can do for AI: a case study, in: *Proceedings of the AIxIA 2020 Discussion Papers Workshop co-located with the the 19th International Conference of the Italian Association for Artificial Intelligence (AIxIA2020)*, volume 2776, CEUR-WS, 2020, pp. 41–48.
- [23] A. Lieto, G. L. Pozzato, A description logic framework for commonsense conceptual combination integrating typicality, probabilities and cognitive heuristics, *Journal of Experimental & Theoretical Artificial Intelligence* 32 (2020) 769–804. doi:[10.1080/0952813X.2019.1672799](https://doi.org/10.1080/0952813X.2019.1672799).
- [24] P. Galliani, O. Kutz, N. Troquard, Perceptron operators that count, in: M. Homola, V. Ryzhikov, R. Schmidt (Eds.), *Proceedings of the 34th International Workshop on Description Logics (DL 2021)*, CEUR Workshop Proceedings, Bratislava, Slovakia, 2021.
- [25] P. Galliani, O. Kutz, N. Troquard, Succinctness and Complexity of  $\mathcal{ALC}$  with Counting Perceptrons, in: *Proceedings of the Twentieth International Conference on Principles of Knowledge Representation and Reasoning (KR 2023)*, Rhodes, Greece, September 2–8, 2023.