Introducing Weighted Prototypes in Description Logics for Defeasible Reasoning

Gabriele Sacco[1,2], Loris Bozzato[1] and Oliver Kutz[2]

[1]Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy

[2]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 study their capability 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 solution is based on the idea of weighted prototypes, a new form of perceptron operator which is used to represent a notion of graded typicality of concept instances.

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 current 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:

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

CILC'23: 38th Italian Conference on Computational Logic, June 21–23, 2023, Udine, Italy

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)

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 pure 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. 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 [5] and its representation based on the *tooth operator* as introduced, for example, in [6]. 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 [7]. 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 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.

We remark that the current presentation of the formalisation 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 base of the idea 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 sketch a proposal for a syntax and semantics of such enriched KBs.

2.1. Syntax

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 Σ with disjoint and non-empty sets NC of concept names, NR of role names, and NI of individual names. Furthermore, we identify a subset of the concept names as denoting *prototype names* by assuming a subset $\mathrm{NP} \subseteq \mathrm{NC}$ and a set of *feature names* $\mathrm{NF} \subseteq \mathrm{NC}$ with $\mathrm{NP} \cap \mathrm{NF} = \emptyset$.

Definition 1 (Features). *A* basic feature *is a concept name* $C \in \mathrm{NF}$*. 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 $\mathrm{NC} \setminus \mathrm{NP} \cup \mathrm{NF}$.

The features associated with prototypes together with the degree of their importance are given in *prototype definitions.*[1] In particular, to allow for a direct comparison across prototype scores, we here constrain the weights of features to be in the $[0, 1]$ interval and to add up to 1, i.e. prototypes are *positive* and *normalised.*

Definition 2 (Positive normalised prototype definition). *Let* $P \in \mathrm{NP}$ *be a prototype name, let* $C_1, \ldots, C_m$ *be general features of* $\mathcal{L}_\Sigma$ *and let* $\overline{w} = (w_1, \ldots, w_m) \in \mathbb{R}^m$ *be a weight vector, where for every* $i \in \{1, \ldots, m\}$ *we have* $w_i \in (0, 1]$ *and* $\sum_{i \in \{1,\ldots,m\}} w_i = 1$*. Then, the expression*

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

is called a prototype definition *for* P*.*

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 $P \in \mathrm{NP}$ and D is a (possibly general) concept of $\mathcal{L}_\Sigma$. Intuitively, these axioms are not absolute and can be "overridden" by prototype instances (cf. defeasible axioms in [8]), 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 admits 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:*

[1]Note that this definition of prototypes is similar to the definition of concepts by the tooth operator defined in [6].

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

$$\mathcal{T} = T_P = \{\, \mathtt{Dog} \sqsubseteq \mathtt{Trusted},\ \mathtt{Wolf} \sqsubseteq \neg\mathtt{Trusted} \,\},$$

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

$$\begin{aligned} \mathcal{P} = \{\, & \mathtt{Wolf}(\mathtt{livesInWoods} : 0.3,\ \mathtt{hasLegs} : 0.1,\ \mathtt{livesInPack} : 0.2,\ \mathtt{Hunts} : 0.4), \\ & \mathtt{Dog}(\mathtt{hasCollar} : 0.3,\ \mathtt{livesInHouse} : 0.2,\ \mathtt{hasLegs} : 0.1,\ \mathtt{isTamed} : 0.4) \,\} \end{aligned}$$

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. 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.* ◇

2.2. Semantics

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* Σ *with a non-empty domain,* $\Delta^\mathcal{I}$*,* $a^\mathcal{I} \in \Delta^\mathcal{I}$ *for every* $a \in \mathrm{NI}$*,* $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$ *for every* $A \in \mathrm{NC}$*,* $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$ *for every* $R \in \mathrm{NR}$*, 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 [8]. Given an axiom $\alpha \in \mathcal{L}_\Sigma$ with FO-translation $\forall \mathbf{x}.\phi_\alpha(\mathbf{x})$, the *instantiation* of α with a tuple $\mathbf{e}$ of individuals in NI, written $\alpha(\mathbf{e})$, is the specialisation of α to $\mathbf{e}$, i.e., $\phi_\alpha(\mathbf{e})$, depending on the type of α.

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

A clashing set *for $\langle \alpha, \mathbf{e} \rangle$ is a satisfiable set S of ABox assertions s.t. $S \cup \{\alpha(\mathbf{e})\}$ 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 χ 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 defeasibly apply prototype axioms in T_P (excluding the exceptional instances in χ).

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 $\alpha \in T_C \cup \mathcal{A}$ 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 $\mathcal{I} \models \phi_\alpha(d)$.

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 $c \in \mathrm{NI}$. 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 $\langle \alpha, \mathbf{e} \rangle \in \chi$ is* justified *for a CAS-model $\mathcal{I}_{CAS}$, if some clashing set $S_{\langle \alpha, \mathbf{e} \rangle}$ 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 $\mathcal{I}' \models S_{\langle \alpha, \mathbf{e} \rangle}$. A CAS model $\mathcal{I}_{CAS}$ of a PKB $\mathfrak{K}$ is* justified, *if every $\langle \alpha, \mathbf{e} \rangle \in \chi$ is justified in $\mathfrak{K}$.*

We define the consequence from justified CAS-models: $\mathfrak{K} \models_{JCAS} \alpha$ 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 member of a prototype is associated with a score which denotes the "degree of typicality" of the instance with respect to the concept described by the prototype. As in [6], 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. Formally, a simple score function can be defined as follows:

Definition 9 (Prototype score). *Given a prototype definition $P(C_1 : w_1, ..., C_m : w_m)$, we define the* score function $score_P : \mathrm{NI} \to [0, 1]$ *for prototype P as:*

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

The 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 $\langle Q \sqsubseteq E, e\rangle \in \chi_2 \setminus \chi_1$, it holds that $score^P_{\mathfrak{K}}(e) < score^Q_{\mathfrak{K}}(e)$.*

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

- *$\mathfrak{K}$ 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 $\mathfrak{K}$ (denoted $\mathfrak{K} \models \alpha$) 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*

$$\chi_1 = \{\langle \texttt{Wolf} \sqsubseteq \neg\texttt{Trusted}, \texttt{balto}\rangle\} \text{ and } \chi_2 = \{\langle \texttt{Dog} \sqsubseteq \texttt{Trusted}, \texttt{balto}\rangle\}.$$

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 $\{\texttt{Wolf}(\texttt{balto}), \texttt{Trusted}(\texttt{balto})\}$ for the clashing assumptions in χ_1 and $\{\texttt{Dog}(\texttt{balto}), \neg\texttt{Trusted}(\texttt{balto})\}$ for that in χ_2, they are both justified. In order to decide which model is preferred, we need to compute the prototype scores for* `balto`*: we have $score_{Wolf}(balto) = 0.4$, $score_{Dog}(balto) = 0.5$, consequently $score_{Wolf}(balto) < score_{Dog}(balto)$ and $\chi_2 > \chi_1$. This means that the preferred model, i.e. the only PKB model, is $\mathcal{I}^1$ where* `balto` *is an exception to* $\texttt{Wolf} \sqsubseteq \neg\texttt{Trusted}$*. Consequently, it holds that* $\mathfrak{K} \models \texttt{Trusted}(\texttt{balto})$.

Moreover, we can note that for `pluto` *and* `alberto` *we can standardly infer* $\texttt{Trusted}(\texttt{pluto})$ *and* $\neg\texttt{Trusted}(\texttt{alberto})$*. 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.* ◇

3. Discussion and Conclusions

We presented an initial formalisation for a non-monotonic extension of DLs with the aim 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 typical it is: in other words, we use typicality to decide with respect 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 that we introduce is not extensional: by using the scores to represent it, we are relying on a characteristic which goes beyond an extensional set-theoretic treatment.

In future work, we want to extend the cognitive and ontological study of exceptions also by comparing it with other accounts for typicality and defeasibility in DLs. Regarding our formalisation, we need to explore and refine the formal consequences of our approach in greater detail. In particular, we need to discuss what are the best options to compute the scores in order to have a balanced score for every prototype and how to extend this computation to roles, possibly following some of the ideas outlined in [9, 10], where novel tooth-operators for role-successor counting are studied. The preference relation can also be refined: for example, comparisons on clashing assumptions can be restricted to the axioms that are actually incompatible. We need also to understand better 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.

Finally, we need an extensive comparison with related works. On the one hand, we will confront our approach to existing formalisms for defeasible reasoning in DLs like [11, 12]. Of particular interest for this purpose are formalisms using weights and having a multi-preferential relation over the individuals with respect to the concepts they are instances of, as, for instance, [13, 14]. On the other hand, we will analyse also works that share our approach to taking into account, in a central way, the results coming from cognitive science and philosophy for developing formal systems in the field of knowledge representation and in particular using the language of DLs. Examples of such works, particularly interested in the notion of typicality, are [15, 16].

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. 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] J. A. Hampton, Concepts as prototypes, volume 46 of *Psychology of Learning and Motivation*, Academic Press, 2006, pp. 79–113.

[6] 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.

[7] 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.

[8] L. Bozzato, T. Eiter, L. Serafini, Enhancing context knowledge repositories with justifiable exceptions, Artif. Intell. 257 (2018) 72–126.

[9] 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.

[10] 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.

[11] 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. doi:https://doi.org/10.1016/j.artint.2015.05.001.

[12] 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). doi:10.1145/3420258.

[13] 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.

[14] 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.

[15] 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.

[16] 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.