# Perceptron Connectives in Knowledge Representation

Pietro Galliani[1], Guendalina Righetti[1], Oliver Kutz[1], Daniele Porello[2], and Nicolas Troquard[1]

[1] Conceptual and Cognitive Modelling Research Group (CORE), KRDB Research Centre for Knowledge and Data, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
[2] Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy.

**Abstract.** We discuss the role of perceptron (or threshold) connectives in the context of Description Logic, and in particular their possible use as a bridge between statistical learning of models from data and logical reasoning over knowledge bases. We prove that such connectives can be added to the language of most forms of Description Logic without increasing the complexity of the corresponding inference problem. We show, with a practical example over the Gene Ontology, how even simple instances of perceptron connectives are expressive enough to represent learned, complex concepts derived from real use cases. This opens up the possibility to import concepts learnt from data into existing ontologies.

**Keywords:** Description Logic · Machine Learning · Perceptrons · Linear Classifiers · Threshold Operators · Ontologies.

## 1 Introduction

*Weighted Threshold Operators* are $n$-ary logical operators which compute a weighted sum of their arguments and verify whether it reaches a certain threshold. These operators have been extensively studied in the context of circuit complexity theory (see e.g. [22]), and they are also known in the neural network community under the alternative name of *perceptrons* (see e.g. [4]).[1]

In [19], threshold operators were studied in the context of Knowledge Representation, focusing in particular on Description Logics (DLs). We refer the reader to [3] for a more thorough introduction to DL. Adding threshold operators to DL is not hard. In brief, if $C_1 \ldots C_n$ are concept expressions, $w_1 \ldots w_n \in \mathbb{R}$ are weights, and $t \in \mathbb{R}$ is a threshold, we can introduce a new concept $\mathbb{W}^t(C_1 : w_1 \ldots C_n : w_n)$ to designate those individuals $d$ such that $\sum\{w_i : C_i \text{ applies to } d\} \geq t$. In the context of DL and concept representation, such

[1] Under the modern understanding of the term, a 'Perceptron' may have an activation function different from the Step Function (in particular, a differentiable one which is more suited to learning via back-propagation in multi-layer networks). In this work, however, we concentrate on the single-layer, step-function case.

threshold ("Tooth") expressions are natural and useful, as they provide a simple way to describe the class of the individuals that satisfy "enough" of a certain set of desiderata. For example, we may wish to state that a student must obtain at least three credits from attending courses **A**, **B**, **C** and **D**, where courses **A** and **B** are worth one credit each and courses **C** and **D** are worth two credits each. This is naturally expressed by the TBox axiom

$$\mathbf{Student} \sqsubseteq \mathbb{W}^{3}(\exists\mathtt{Att}.\mathbf{A} : 1, \exists\mathtt{Att}.\mathbf{B} : 1, \exists\mathtt{Att}.\mathbf{C} : 2, \exists\mathtt{Att}.\mathbf{D} : 2)$$

where `Att` represents the "attends" role. Suppose now that course **A** became compulsory. This could be done in two distinct ways: we could explicitly demand that students attend course **A**, thus turning the above axiom into

$$\mathbf{Student} \sqsubseteq (\exists\mathtt{Att}.\mathbf{A}) \sqcap \mathbb{W}^{3}(\exists\mathtt{Att}.\mathbf{A} : 1, \exists\mathtt{Att}.\mathbf{B} : 1, \exists\mathtt{Att}.\mathbf{C} : 2, \exists\mathtt{Att}.\mathbf{D} : 2),$$

or we could simply assign more credits to course **A** and increase the credits requirement, thus turning the above axiom to e.g.

$$\mathbf{Student} \sqsubseteq \mathbb{W}^{12}(\exists\mathtt{Att}.\mathbf{A} : 10, \exists\mathtt{Att}.\mathbf{B} : 1, \exists\mathtt{Att}.\mathbf{C} : 2, \exists\mathtt{Att}.\mathbf{D} : 2).$$

These last two possibilities are semantically equivalent: in either scenario, a student has to attend course **A** and at least two of the others. However, they convey subtly different situations, and would lead to different consequences should the ontology be modified further (e.g. by adding another course **E** that is worth 10 credits by itself).

For a less mundane example, consider the *Felony Score Sheet* used in the State of Florida[2], in which various aspects of a crime are assigned points, and a threshold must be reached to decide compulsory imprisonment. For example, possession of cocaine corresponds to 16 points if it is the primary offense and to 2.4 points otherwise, a victim injury describable as "moderate" corresponds to 18 points, and a failure to appear for a criminal proceeding results in 4 points. Imprisonment is compulsory if the total is greater than 44 points and not compulsory otherwise. A knowledge base describing the laws of Florida would need to represent this score sheet as part of its definition of its **CompulsoryImprisonment** concept, for instance as

$$\mathbb{W}^{44}(\mathbf{CocainePrimary} : 16, \mathbf{ModerateInjuries} : 18, \ldots).$$

While it would be possible to also describe it (or any other Boolean function) in terms of more ordinary logical connectives (e.g. by a DNF expression), a definition in terms of Tooth expressions is far simpler and more readable. As such, the definition is more transparent and more explainable.

We refer the interested reader to [19] and to [13] for a more in-depth analysis of the properties of this operator. The paper [13] also introduces a *knowledge-dependent* variant of the threshold operator, in which the individuals are not

[2] http://www.dc.state.fl.us/pub/scoresheet/cpc_manual.pdf (accessed: 20 May 2020)

scored with respect to the current interpretation but with respect to some knowledge base $\mathcal{K}$ (which, in the case of a felony score sheet, may describe for example the findings according to which the score is to be computed). Having Tooth expressions in a language of knowledge representation has notable advantages, from a cognitive point of view and from the practical point of view of knowledge acquisition. First, in psychology and cognitive science, the combination of two or more concepts has a more subtle semantics than set theoretic operations. As shown in [20], Tooth operators can be used to represent these new concepts more faithfully regarding the way in which humans think of them and combine them. Second, as illustrated in [13], since a Tooth expression is simply a linear classification model, it is possible to use standard linear classification algorithms (such as the Perceptron Algorithm, Logistic Regression, or Linear SVM) to learn its weights and its threshold given a set of assertions about individuals (that is, given an ABox).

Extensions of Description Logic involving threshold operators have also been discussed in [1] and [2]. The approaches presented in these two papers are, however, very different from the one summarized above: the former paper, indeed, changes the semantics of Description Logic by associating *graded membership functions* to models and requiring them for the interpretation of expressions, while the latter one extends the semantics of the Description Logic $\mathcal{ALC}$ by means of weighted alternating parity tree automata. The approach described above is, in comparison, more direct: no changes are made to the definitions of the models of the Description Logic(s) to which threshold operators are added, and the language is merely extended by means of the above-described operators, which as already pointed out in [19], can be easily seen not to increase the expressive power of any language that contains the ordinary Boolean operators.

Aside from these technical differences, we argue that the approach introduced in [19] is more adequate even from a cognitive point of view. Although the proposal of Baader *et al.* allows one to represent concepts in an approximate way, introducing weights in the language permits to represent in a more straightforward way the relative importances of the different features participating in the concept descriptions. Tooth operators are in fact in line with the classical definition of prototypes given in the Prototype Theory exploited in the cognitive sciences (see e.g. [17, chapter 3]). Moreover, threshold expressions of [19] are putatively more intuitive and readable for non logic-experts, making them more cognitively adequate and less error-prone.

Two questions, however, need to be answered in order to assess the viability of this proposed addition to the language(s) of Description Logic:

1. Given a Description Logic $\mathcal{L}$, let $\mathcal{L}(\mathbb{W})$ be the logic obtained by adding threshold operators to it. How does the inference problem for $\mathcal{L}(\mathbb{W})$ compare to that for $\mathcal{L}$? More specifically: let $\mathcal{K}$ be a $\mathcal{L}(\mathbb{W})$-knowledge base and let $\phi$ be a $\mathcal{L}(\mathbb{W})$ axiom. Can we reduce the problem of whether $\mathcal{K} \models \phi$ (that is, of whether every interpretation that satisfies $\mathcal{K}$ satisfies $\phi$) to the problem of whether $\mathcal{K}_0 \models \phi_0$ for some $\mathcal{K}_0, \phi_0 \in \mathcal{L}$ with an at-most-polynomial overhead?

2. Can we find examples in which simple threshold expressions can be used to express, more shortly and readably than (but roughly as accurately as) alternative approaches, non-trivial concepts derived from real data? If so, this would validate the claim that such expressions are well-suited for representing complex concepts in a readable way [19,20].

In what follows, we will answer these two questions.

## 2 Translating Threshold Expressions

The key ingredient for our result will be the following Proposition:

**Proposition 1.** *Let* $\mathtt{T} = \mathbb{W}^t(C_1 : w_1 \dots C_n : w_n)$ *be any* $\mathcal{L}(\mathbb{W})$ *threshold expression, where* $C_1 \dots C_n$ *are* $\mathcal{L}$*-concepts and* $t, w_1 \dots w_n$ *are positive integers. Furthermore, let* $\mathtt{TOOTH}$ *be an atomic concept symbol not appearing in* $\mathtt{T}$.

*Then we can build a knowledge base* $\mathcal{K}(\mathtt{T} \mapsto \mathtt{TOOTH})$ *in* $\mathcal{L}$*, containing expressions built out of the concepts expressions* $C_1 \dots C_n$ *and of a number of fresh atomic symbols (including* $\mathtt{TOOTH}$*) such that*

1. $\mathcal{K}(\mathtt{T} \mapsto \mathtt{TOOTH}) \models \mathtt{TOOTH} \equiv \mathtt{T}$*;*
2. *Every interpretation* $I$ *whose signature contains the atoms contained in* $\mathtt{T}$ *but not the fresh atoms introduced by* $\mathcal{K}(\mathtt{T} \mapsto \mathtt{TOOTH})$ *can be expanded in one and only one way into some* $I'$ *that satisfies* $\mathcal{K}(\mathtt{T} \mapsto \mathtt{TOOTH})$*;*
3. *The size of* $\mathcal{K}(\mathtt{T} \mapsto \mathtt{TOOTH})$ *is polynomial in the size of* $\mathtt{T}$.[3]

Before proving this, let us show that it leads to the intended conclusion. A consequence of Proposition 1 is the following:

**Proposition 2.** *Let* $C$ *be any* $\mathcal{L}(\mathbb{W})$*-concept. Then we can find an* $\mathcal{L}$*-theory* $\mathcal{K}_C$*, of size polynomial in the size of* $C$ *and containing the symbols occurring in* $C$ *as well as a number of fresh atomic concept symbols, and a* $\mathcal{L}$ *concept expression* $C'$ *of size smaller or equal than that of* $C$*, such that*

1. $\mathcal{K}_C \models C \equiv C'$*;*
2. *Every interpretation* $I$ *whose signature contains the symbols of* $C$ *but not the fresh symbols added by* $\mathcal{K}_C$ *can be expanded in one and only one way to an interpretation* $I'$ *that satisfies* $\mathcal{K}_C$*.*

Then the desired theorem follows at once:

**Theorem 1.** *Let* $\mathcal{L}$ *be a Description Logic that contains all Boolean connectives, let* $\mathcal{K}$ *be a* $\mathcal{L}(\mathbb{W})$ *knowledge base and let* $\phi$ *be a* $\mathcal{L}(\mathbb{W})$ *axiom. Then, the problem of whether* $\mathcal{K} \models \phi$ *can be reduced, with polynomial overhead, to the problem of whether* $\mathcal{K}_L \models \phi_L$ *for some* $\mathcal{L}$ *knowledge base* $\mathcal{K}_L$ *and some* $\mathcal{L}$ *axiom* $\phi_L$*.*

[3] For the purposes of this work, the size of a concept expression includes also the number of bits required to express the weights and thresholds eventually occurring in it.

Thus, the inference problem in $\mathcal{L}(\mathbb{W})$ can indeed be reduced efficiently to the inference problem in $\mathcal{L}$ whenever Boolean connectives are already in the language of $\mathcal{L}$.

It remains to verify that Proposition 1 holds. So let $\mathbb{W}^t(C_1 : w_1 \dots C_n : w_n)$ be our threshold expression, let $k$ be the number of binary digits required to write the threshold and the (positive) weights, and let us also assume the other premises of Proposition 1. What we ultimately will do is writing the specification of a ripple-carry adder[4] and of a digital number comparator in the syntax of Description Logic.

For the sake of clarity, we will do so in several steps:

### 2.1 Encoding the weights

Let $W_{ij} : i \in 1 \dots n, j \in 0 \dots k-1$ and $T_j : j \in 0 \dots k-1$ be fresh atoms. Then let $\mathcal{K}_0$ be the TBox containing

- $W_{ij} \equiv C_i$, for all $i \in 1 \dots n$ and for all $j \in 0 \dots k-1$ such that the $j$-th least significant digit of the binary representation of $w_i$ is 1, and $W_{ij} \equiv \bot$ for all the others;
- $T_j \equiv \top$ for all $j \in 0 \dots k-1$ such that the $j$-th least significant digit of the binary representation of $t$ is 1, and $T_j \equiv \bot$ for the others.

**Lemma 1.** *$\mathcal{K}_0$ has size polynomial in the size of our original threshold expression. Moreover, any interpretation $I$ in which $C_1 \dots C_n$ can be interpreted and in which the fresh atoms $W_{ij}$ and $T_j$ do not appear has a* unique *extension to an interpretation $I'$ such that $I' \models \mathcal{K}_0$. For that interpretation, we furthermore have that, for all individuals $d \in \Delta^{I'}$,*

$$\sum\{2^j : j = 0 \dots k-1, d \in W_{ij}^{I'}\} = \begin{cases} w_i & \text{if } d \in C_i^I; \\ 0 & \text{otherwise.} \end{cases}$$

*for all $i \in 1 \dots n$. Likewise,*

$$\sum\{2^j : j = 0 \dots k-1, d \in T_j^{I'}\} = t.$$

### 2.2 Encoding the sum

**Summing the first weight** We define[5] $\mathcal{K}_1$ as the union of $\mathcal{K}_0$ and the following axioms, for the fresh atomic symbols $\mathtt{SUM}^1_0 \dots \mathtt{SUM}^1_{k-1}$:

- For all $j = 0 \dots k-1$, we add the axiom $\mathtt{SUM}^1_j \equiv W_{1j}$.

[4] Other, more efficient types of adder circuits are known and used in practice, and could be translated along similar lines; but ripple-carry adders have the advantage of simplicity and suffice for our purposes.

[5] This first sum is strictly speaking unnecessary, but we keep it for clarity of exposition.

**Lemma 2.** *$\mathcal{K}_1$ has size polynomial in the size of our original threshold expression. Let $I$ be as in Lemma 1: then $I$ has exactly one expansion to a model $I'$ of $\mathcal{K}_1$, and for $I'$ we have that $\sum\{2^j : j = 0 \ldots k-1, d \in (\mathtt{SUM}^1_j)^{I'}\} = \sum\{w_i : 1 \leq i \leq 1, d \in C_i^I\}$.*

**Summing the other weights** For $i = 2 \ldots n$, we define inductively $\mathcal{K}_i$ as $\mathcal{K}_{i-1}$ plus the following axioms (for fresh symbols $\mathtt{SUM}^i_0 \ldots \mathtt{SUM}^i_{k-1}$ and $\mathtt{CARRY}^i_0 \ldots \mathtt{CARRY}^i_{k-1}$) and $\mathtt{OVERFLOW}^i$:

- The axiom $\mathtt{CARRY}^i_0 \equiv \bot$;
- For all $j = 0 \ldots k-1$, the axiom

$$\begin{aligned}\mathtt{SUM}^i_j \equiv &(\mathtt{CARRY}^i_j \sqcap \mathtt{SUM}^{i-1}_j \sqcap W^i_j) \sqcup (\mathtt{CARRY}^i_j \sqcap \neg\mathtt{SUM}^{i-1}_j \sqcap \neg W^i_j) \sqcup \\ &(\neg\mathtt{CARRY}^i_j \sqcap \mathtt{SUM}^{i-1}_j \sqcap \neg W^i_j) \sqcup (\neg\mathtt{CARRY}^i_j \sqcap \neg\mathtt{SUM}^{i-1}_j \sqcap W^i_j);\end{aligned}$$

- For all $j = 1 \ldots k-1$, the axiom

$$\mathtt{CARRY}^i_j \equiv (\mathtt{CARRY}^i_{j-1} \sqcap \mathtt{SUM}^{i-1}_{j-1}) \sqcup (\mathtt{CARRY}^i_{j-1} \sqcap W^i_{j-1}) \sqcup (\mathtt{SUM}^{i-1}_{j-1} \sqcap W^i_{j-1});$$

- The axiom

$$\mathtt{OVERFLOW}^i \equiv (\mathtt{CARRY}^i_{k-1} \sqcap \mathtt{SUM}^{i-1}_{k-1}) \sqcup (\mathtt{CARRY}^i_{k-1} \sqcap W^i_{k-1}) \sqcup (\mathtt{SUM}^{i-1}_{k-1} \sqcap W^i_{k-1}).$$

**Lemma 3.** *For all $\ell = 1 \ldots n$, $\mathcal{K}_\ell$ has size polynomial in the size of our original threshold expression.*

*Moreover, for every such $\ell$, every interpretation $I$ as in Lemma 1 can be extended in exactly one way to an interpretation $I'$ which satisfies $\mathcal{K}_\ell$; and for this interpretation $\mathtt{SUM}^\ell_{k-1} \ldots \mathtt{SUM}^\ell_0$ is a binary encoding of the sum of the weights (up to $w_\ell$) which correspond to concepts that apply to the current individual, in the sense that (for all $d \in \Delta^{I'}$) $\sum\{2^j : j = 0 \ldots k-1, d \in (\mathtt{SUM}^\ell_j)^{I'}\} = \sum\{w_i : 1 \leq i \leq \ell, d \in C^I\}$ whenever that value is less than $2^k$, and $d \in (\mathtt{OVERFLOW}^\ell)^{I'}$ otherwise.*

*In particular, if $I \models \mathcal{K}_n$ then $\sum\{2^j : j = 0 \ldots k-1, d \in (\mathtt{SUM}^n_j)^{I'}\} = \sum\{w_i : 1 \leq i \leq n, d \in C^I\} = v^I_{\mathtt{T}}(d)$ is the value of our tooth expression $\mathtt{T} = \nabla^t(C_1 : w_1 \ldots C_n : w_n)$ if that value is less than $2^k$, and otherwise $d \in (\mathtt{OVERFLOW}^i)^{I'}$ for at least one $i = 2 \ldots n$.*

### 2.3 Comparing with the threshold

Now define $\mathcal{K}$ as $\mathcal{K}_n$ plus the following axioms (for fresh atoms $\mathtt{EQ}_{k-1} \ldots \mathtt{EQ}_0$, $\mathtt{MAJ}_{k-1} \ldots \mathtt{MAJ}_0$, $\mathtt{TOOTH}$:

- $\mathtt{EQ}_{k-1} \equiv ((\mathtt{SUM}^n_{k-1} \sqcap T_{k-1}) \sqcup (\neg\mathtt{SUM}^n_{k-1} \sqcap \neg T_{k-1}))$;
- For $j = (k-2) \ldots 0$, the axiom

$$\mathtt{EQ}_j \equiv \mathtt{EQ}_{j+1} \sqcap ((\mathtt{SUM}^n_j \sqcap T_j) \sqcup (\neg\mathtt{SUM}^n_j \sqcap \neg T_j));$$

- $\mathtt{MAJ}_{k-1} \equiv \mathtt{SUM}^n_{k-1} \sqcap \neg T_{k-1}$;
- For $j = (k-2) \dots 0$, the axiom

$$\mathtt{MAJ}_j \equiv \mathtt{EQ}_{j+1} \sqcap \mathtt{SUM}^n_j \sqcap \neg T_j;$$

- The axiom

$$\begin{aligned} \mathtt{TOOTH} \equiv & \mathtt{OVERFLOW}^2 \sqcup \dots \sqcup \mathtt{OVERFLOW}^n \sqcup \\ & \mathtt{MAJ}_{k-1} \sqcup \dots \sqcup \mathtt{MAJ}_0 \sqcup \mathtt{EQ}_0. \end{aligned}$$

**Lemma 4.** *$\mathcal{K}$ has size polynomial in the size of our original threshold expression. Moreover, every interpretation $I$ as in Lemma 1 can be extended in exactly one way to an interpretation $I'$ that satisfies $\mathcal{K}$; and for this interpretation and for every individual $d \in \Delta^{I'}$,*

- *For all $j = k-1 \dots 0$, $d \in \mathtt{EQ}^I_j$ if and only if the binary encodings of $v^I_{\mathtt{T}}(d)$ and of $t$ agree from the most significant digit to the $j$-th least significant digit;*
- *For all $j = k-1 \dots 0$, $d \in \mathtt{MAJ}^I_j$ if and only if the binary encodings of $v^I_{\mathtt{T}}(d)$ and of $t$ disagree on the $j$-th least significant digit, which is greater for $v^I_{\mathtt{C}}(d)$ than for $t$, but agree on all the digits on the left of it;*
- *$d \in \mathtt{TOOTH}^{I'}$ if and only if we obtained an overflow when summing all the weights which apply to the individual $d$ (remember that we assumed positive weights, so this implies at once that $v^I_{\mathtt{T}}(d)$ is greater than the threshold), or if there is a digit that is greater for $v^I_{\mathtt{R}}(d)$ than for $t$ and all the digits to the left agree, or if all the digits of $v^I_{\mathtt{T}}(d)$ and of $t$ are the same - that is, if and only if $v^I_{\mathtt{T}}(d) \geq t$.*

At this point, Proposition 1 follows at once by picking this $\mathcal{K}$ for $\mathcal{K}(\mathtt{T} \mapsto \mathtt{TOOTH})$.

## 3 Learning Simple Threshold Expressions

In order to evaluate the practical usefulness of threshold expressions, we are going to investigate whether simple non-nested threshold expressions suffice to represent adequately Gene Ontology concepts.

*The Gene Ontology.* The Gene Ontology [12] (GO) is a knowledge base consisting (by January 2020) of 44,700 different concepts ("terms") annotating more than one million gene products from 4,591 different species. Different concepts relate to each other not only via the usual subsumption ("is-a") relation, but also via other relations such as "part-of" or "regulates"; and they are partitioned into the three disjoint sub-ontologies of *Cellular Component*, for concepts relating to locations inside of a cell, like "nucleus" (term GO:0005634) or "Golgi Apparatus" (term GO:0005794); *Biological Process*, for concepts specifying "biological programs" to which a gene product participates, like "Asexual Reproduction" (term GO:0019954) or "Oxygen Transport" (term GO:0015671); and *Molecular Function*, for concepts relative to specific molecular-level roles performed by gene products such as “Enzyme Binding” (term GO:0019899) or “Structural Constituent of Ribosome” (term GO:0003735).

Datasets exist that associate gene products to Gene Ontology terms: for example, according to the Saccharomyces Genome Database [9,10],[6] the enzyme *ATP synthase* (ATP8) is located in the *mitochondrion* (GO:0005739), is involved in the biological processes of *ion transport* (GO:0006811) and *transmembrane transport* (GO:0055085), and more specifically *ATPase activity* (GO:0016887) and *hydrolase activity* (GO:0016787) are among its molecular functions.

*Approach.* For the purposes of this work, we decided to focus on the annotations of the Saccharomyces Genome Database and on the subset of the Gene Ontology (the “GO slim”, in the terminology used by the Gene Ontology Consortium) that has been curated by it for the purpose of annotating yeast gene products. We likewise downloaded Gene Ontology annotations of yeast gene products from the website of the Saccharomyces Genome Database. Then we considered the following question: up to which degree is it possible to infer the Molecular Function annotations of a gene product from its Cellular Component and Biological Process ones? In other words, given the locations of a gene product inside of a yeast cell and the overall “cellular programs” it is involved in, can we infer (to some degree, at least) its specific molecular-level roles?

It is worth emphasizing here that our purpose is *not* to design and propose a novel state-of-the-art machine learning algorithm. Rather, our aim is to investigate the expressive potential of simple threshold expressions in real use scenarios. To this purpose, we designed a very basic evolutionary algorithm to extract threshold expressions from data. In brief, a population of one hundred random threshold expressions (with Gene Ontology concepts as arguments, integer weights, at most 10 arguments, and threshold fixed at 100) is generated, then they attempt to “copy” (concept, weight) pairs from randomly selected neighbours[7], keeping them if they improve the performance on the training data; weights are mutated randomly, and the mutation is likewise kept if it is an improvement; and every ten turns the half worst-performing threshold expressions are removed and replaced with random ones. After one thousand turns, we simply return the threshold expression that performs best over the training data. This is only a cursory description, but again, we wish to make it clear that this algorithm is merely a means to an end—the end being to verify whether simple threshold expressions can adequately capture complex concepts. No serious attempt was made to fine-tune its performance or refine its overall design.

As a baseline, we used a few state-of-the-art learning algorithms as implemented in the Waikato Environment for Knowledge Analysis (WEKA) [14], namely a Random Forest classifier [7], the Sequential Minimal Optimization

[6] The Saccharomyces Genome Database is available at https://www.yeastgenome.org/.

[7] While maintaining the maximum number of components of every threshold expression to 10: if that number has been reached, the copied weight replaces the component whose weight has the smallest magnitude.

algorithm for Support Vector Machines [18], a decision table majority classifier [15], a logistic regression classifier [8] and a multilayer perceptron classifier [21].

Since the available data is heavily imbalanced (for every possible molecular function, most gene products will not be associated with it), we decided to use Matthews Correlation as our performance measure [16,6,11], which describes the statistical correlation between the predicted label and the true one and is 0 if these are uncorrelated and 1 if there is perfect positive correlation. As discussed, e.g., in [11], other standard metrics for classification such as accuracy or the $F_1$ score can lead to overoptimistic results on greatly imbalanced datasets. The Matthews Correlation Coefficient makes use of all four cells of the confusion matrix (true/false, positives/negatives), and it assigns the same importance to "positive" and "negative" examples. It has a natural statistical interpretation: it can be seen as the special case of the Pearson Correlation Coefficient in which variables may only take one of two values.

*Data preparation.* We prepared the data as follows: first, we removed all gene product annotations listed as "dubious" in the Saccharomyces Genome Database, as well as the annotations to the three uninformative top-level terms of the three sub-ontologies. Then we picked from the mapping file of the Saccharomyces Genome Database gene products with at least three annotations of type "Cellular Component" or "Biological Process". Then we chose as the labels to predict the "Molecular Function" type annotations that occur in at least one hundred of the above gene products, and selected as features the "Cellular Component" or "Biological Process" terms that apply to at least one of these gene products. This resulted in a dataset of 4,595 gene products, each one of which has 120 features and 17 possible labels. For each of these labels, we split the gene products in five folds, maintaining the same proportions of true labels.

We reserved five of these labels for final testing, using the others for developing our approach to learning threshold expressions and for tuning our baselines. This highlighted in particular that, in the cases of Decision Majority Tables, Logistic Classifiers and Support Vector Machines, it was necessary to correct the unbalancedness of the data by oversampling the positive examples during training.

*Evaluation.* Finally, we tested our approach on the reserved labels and the corresponding datasets, training our method and our baselines—for each of the five labels in the reserved dataset—on four folds and testing it on the remaining one. The results are summarized in Table 1. The performance varies between labels and for some of them it is not very high, which was only to be expected since in general the biological processes to which a gene product participates and the cellular components in which it is found are not adequate information to infer their molecular function; but what is of interest for our purposes is that the performance of threshold expressions (despite the very basic approach that we took to their learning) follows roughly that of our baselines and is overall as good as them. This supports our hypothesis that threshold expressions, and very simple ones at that, can adequately capture complex concepts in real world scenarios

| | RF | SVM | DT | LR | MLP | $\mathbb{W}$ |
|---|---|---|---|---|---|---|
| GO:0016787 | .34 (.02) | .30 (.03) | .22 (.03) | .30 (.03) | .26 (.07) | .22 (.06) |
| GO:0016301 | .67 (.07) | .53 (.06) | .51 (.09) | .66 (.06) | .79 (.03) | .75 (.04) |
| GO:0030234 | .25 (.06) | .18 (.01) | .12 (.03) | .20 (.04) | .22 (.07) | .27 (.06) |
| GO:0022857 | .80 (.02) | .71 (.04) | .55 (.02) | .79 (.02) | .75 (.03) | .72 (.05) |
| GO:0016740 | .50 (.01) | .48 (.03) | .47 (.04) | .45 (.04) | .48 (.02) | .47 (.03) |

**Table 1.** Matthews Correlations of predictions on five Molecular Function terms. We report averages between five folds and standard deviation. (Leading zeros are omitted.) RF = Random Forest, SVM = Support Vector Machine, DT = Decision Table, LR = Logistic Regression, MLP = Multilayer Perceptron, $\mathbb{W}$ = our Threshold Expressions. The five rows correspond to the Molecular Function Gene Ontology terms GO:0016787 (hydrolase activity), GO:0016301 (kinase activity), GO:0030234 (enzyme regulator activity), GO:0022857 (transmembrane transporter activity) and GO:0016740 (transferase activity).

as well as more sophisticated models despite being of simpler understanding and (as we saw) easier to integrate with logical reasoning.

## 4 Conclusions

The results of this work lend support to the feasibility of adding threshold connectives to knowledge representation languages. As showed in Section 2, such connectives can be added to the language of any DL that has all Boolean connectives without increasing the complexity of the corresponding inference problem, and thus reasoning services for any such DL $\mathcal{L}$ can be also used (after translation) also for the corresponding extension $\mathcal{L}(\mathbb{W})$.

Furthermore, as we showed in Section 3 with a practical example over the Gene Ontology, even simple instances of perceptron connectives are expressive enough to represent complex notions in real use cases.

Much more can be done. A particularly intriguing aspect is the experimental evaluation of the degree up to which threshold expressions are more human-interpretable than equivalent logical formulations, along the lines of [5]. Also, with the prospect of sharing ontologies with perceptron connectives, their addition to semantic web languages will need to be carefully pursued.

## References

1. Baader, F., Brewka, G., Gil, O.F.: Adding threshold concepts to the description logic $\mathcal{EL}$. In: Lutz, C., Ranise, S. (eds.) Frontiers of Combining Systems. pp. 33–48. Springer International Publishing, Cham (2015)
2. Baader, F., Ecke, A.: Reasoning with prototypes in the description logic $\mathcal{ALC}$ using weighted tree automata. In: Dediu, A.H., Janoušek, J., Martín-Vide, C., Truthe, B. (eds.) Language and Automata Theory and Applications. pp. 63–75. Springer International Publishing, Cham (2016)

3. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: Introduction to Description Logic. Cambridge University Press (2017)
4. Bishop, C.M.: Pattern recognition and machine learning. Springer Science+ Business Media (2006)
5. Booth, S., Muise, C., Shah, J.: Evaluating the interpretability of the knowledge compilation map: Communicating logical statements effectively. Proc. 28th IJCAI pp. 5801–5807 (2019)
6. Boughorbel, S., Jarray, F., El-Anbari, M.: Optimal classifier for imbalanced data using matthews correlation coefficient metric. PloS one **12**(6), e0177678 (2017)
7. Breiman, L.: Random forests. Machine learning **45**(1), 5–32 (2001)
8. le Cessie, S., van Houwelingen, J.: Ridge estimators in logistic regression. Applied Statistics **41**(1), 191–201 (1992)
9. Cherry, J.M., Adler, C., Ball, C., Chervitz, S.A., Dwight, S.S., Hester, E.T., Jia, Y., Juvik, G., Roe, T., Schroeder, M., et al.: Sgd: Saccharomyces genome database. Nucleic acids research **26**(1), 73–79 (1998)
10. Cherry, J.M., Hong, E.L., Amundsen, C., Balakrishnan, R., Binkley, G., Chan, E.T., Christie, K.R., Costanzo, M.C., Dwight, S.S., Engel, S.R., et al.: Saccharomyces genome database: the genomics resource of budding yeast. Nucleic acids research **40**(D1), D700–D705 (2012)
11. Chicco, D., Jurman, G.: The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. BMC genomics **21**(1), 6 (2020)
12. Consortium, G.O.: The Gene Ontology (GO) database and informatics resource. Nucleic acids research **32**(suppl_1), D258–D261 (2004)
13. Galliani, P., Kutz, O., Porello, D., Righetti, G., Troquard, N.: On knowledge dependence in weighted description logic. In: Proc. of the 5th Global Conference on Artificial Intelligence (GCAI 2019). pp. 17–19 (2019)
14. Holmes, G., Donkin, A., Witten, I.H.: Weka: A machine learning workbench. In: Proceedings of ANZIIS'94-Australian New Zealnd Intelligent Information Systems Conference. pp. 357–361. IEEE (1994)
15. Kohavi, R.: The power of decision tables. In: European conference on machine learning. pp. 174–189. Springer (1995)
16. Matthews, B.W.: Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et Biophysica Acta (BBA)-Protein Structure **405**(2), 442–451 (1975)
17. Murphy, G.: The Big Book of Concepts. MA: The MIT Press, Cambridge (2002)
18. Platt, J.: Fast training of support vector machines using sequential minimal optimization. advances in kernel methods—support vector learning (pp. 185–208). AJ, MIT Press, Cambridge, MA (1999)
19. Porello, D., Kutz, O., Righetti, G., Troquard, N., Galliani, P., Masolo, C.: A toothful of concepts: Towards a theory of weighted concept combination. In: Proceedings of the 32nd International Workshop on Description Logics. vol. 2373. CEUR-WS (2019), http://ceur-ws.org/Vol-2373/paper-24.pdf
20. Righetti, G., Porello, D., Kutz, O., Troquard, N., Masolo, C.: Pink panthers and toothless tigers: Three problems in classification. In: Proc. of the 5th International Workshop on Artificial Intelligence and Cognition. Manchester, September 10–11 (2019)
21. Rosenblatt, F.: Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Tech. rep., Cornell Aeronautical Lab Inc Buffalo NY (1961)
22. Vollmer, H.: Introduction to circuit complexity: a uniform approach. Springer Science & Business Media (2013)