

# Threshold Operators in Knowledge Representation

Pietro Galliani<sup>1</sup>   Oliver Kutz<sup>1</sup>   Daniele Porello<sup>2</sup>  
Guendalina Righetti<sup>1</sup>   Nicolas Troquard<sup>1</sup>

1. Free University of Bozen-Bolzano (UNIBZ)

2. Laboratory for Applied Ontology (ISTC-CNR, Trento)

## Example

- Course A: 1 credit
- Course B: 1 credit
- Course C: 2 credits
- Course D: 2 credits
- A student must gain at least 3 credits.

$$\text{Student} \sqsubseteq (\mathbf{C} \cap \mathbf{D}) \cup ((\mathbf{A} \cup \mathbf{B}) \cap (\mathbf{C} \cup \mathbf{D}))$$

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

Semantically equivalent, but more **human-readable** and more **easy to update**.

## Example

- Course A: 1 credit
- Course B: 1 credit
- Course C: 2 credits
- Course D: 2 credits
- A student must gain at least 3 credits.

$$\text{Student} \sqsubseteq (\mathbf{C} \cap \mathbf{D}) \sqcup ((\mathbf{A} \sqcup \mathbf{B}) \cap (\mathbf{C} \sqcup \mathbf{D}))$$

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

Semantically equivalent, but more **human-readable** and more **easy to update**.

## Example

- Course A: 1 credit
- Course B: 1 credit
- Course C: 2 credits
- Course D: 2 credits
- A student must gain at least 3 credits.

$$\text{Student} \sqsubseteq (\mathbf{C} \cap \mathbf{D}) \sqcup ((\mathbf{A} \sqcup \mathbf{B}) \cap (\mathbf{C} \sqcup \mathbf{D}))$$

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

Semantically equivalent, but more **human-readable** and more **easy to update**.

## Example (cont,)

- Course A: 1 credit
- Course B: 1 credit
- Course C: 1 credit
- Course D: 2 credits
- A student must gain at least 3 credits.

$$\text{Student} \sqsubseteq (\mathbf{A} \cap \mathbf{B} \cap \mathbf{C}) \cup ((\mathbf{A} \cup \mathbf{B} \cup \mathbf{C}) \cap \mathbf{D})$$

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

## Example (Florida Felony Score Sheet)

- Possession of Cocaine: +16 “felony points”;
- Moderate injuries: +18 “felony points”;
- Failure to appear: +4 “felony points”;
- ...

If total  $\geq 44$ , imprisonment is compulsory; otherwise, not so.



How to express COMPULSORY\_IMPRISONMENT in a knowledge base? Certainly possible (e.g. Disjunctive Normal Form), but not very short or readable...

$$\text{COMPULSORY\_IMPRISONMENT} \equiv (\text{COCAINE} \sqcap \text{FAILURE\_TO\_APPEAR} \sqcap \dots) \sqcup \dots$$

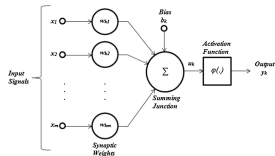
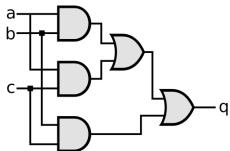
With threshold operators, much clearer:

$$\text{COMPULSORY\_IMPRISONMENT} \equiv \nabla^{44}(\text{COCAINE} : 16, \text{MODERATE\_INJURIES} : 18, \dots)$$

## Links with complexity theory and with learning

**Weighted Threshold Operators** have been studied in the context of propositional logic and **circuit complexity**: [Valiant 1984], [Hajnal et al. 93], [Beimel and Weinreb 2006], [Goldmann et al. 1992] [Goldmann and Karpinski 1998].

In that context, these operators have been seen to be closely related important open problems in **complexity theory**.



Here, instead, we are interested in their possible application to **Knowledge Representation** in Description Logic.

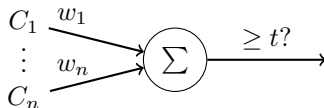
Tooth operators are simple, conceptually clear connectives that—because of their obvious connections with linear classification models—provide a natural link between knowledge representation and **statistical learning**.

## What We Propose

Extend languages for the representation of knowledge (e.g. DLs) with **Weighted Threshold Operators**

$$(\mathbb{W}^t(C_1 : w_1, \dots, C_n : w_n))^I = \left\{ d \in \Delta^I : \sum_{d \in C_k^I} w_k \geq t \right\}$$

- Does not increase expressive power (if usual Boolean connectives available);
- More *human-understandable* in many context;
- Easier to *update*;
- Easier to *learn* (linear classifier!).





## Two Important Questions

- 1 How costly is reasoning with threshold? Given a knowledge base  $K \in \mathbf{L}(\mathbb{V})$  and a  $\phi \in \mathbf{L}(\mathbb{V})$ , how hard is it to decide whether  $K \models \phi$  compared to the same problem without  $\mathbb{V}$ ?
- 2 Are small, human-readable threshold expressions powerful enough to learn and express useful concepts?

## Two Answers

- 1 How costly is reasoning with threshold? Given a knowledge base  $K \in \mathbf{L}(\nabla)$  and a  $\phi \in \mathbf{L}(\nabla)$ , how hard is it to decide whether  $K \models \phi$  compared to the same problem without  $\nabla$ ?
  - ▶ (If  $\mathbf{L}$  contains Boolean connectives): not harder than the inference problem for  $\mathbf{L}$ , up to polynomial reduction.
- 2 Are small, human-readable threshold expressions powerful enough to learn and express useful concepts?
  - ▶ YES!

## Our Claim

If a language for the representation of knowledge has the usual Boolean connectives but cannot represent directly threshold expressions, **it probably should**.

- + Threshold expressions appear often in practical applications;
  - + Threshold expressions are easily learned from data;
  - + Threshold expressions are easily human-readable (if not big);
  - + Threshold expressions can be reduced away easily.
- ???

Threshold expressions are just *syntactic sugar*? In a way, but it's **useful syntactic sugar** and it should be readily available in most knowledge representation languages.

## Language

For  $A \in N_C$ ,  $R \in N_R$ ,  $t \in \mathbb{Z}$ ,  $m \in \mathbb{Z}_{>0}$ ,  $\vec{w} \in \mathbb{Z}^m$  the set of  $\mathcal{ALC}(\nabla)$  concepts is described by the grammar:

$$C ::= \perp \mid \top \mid A \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C \mid \\ \nabla^t(C_1 : w_1, \dots, C_m : w_m)$$

(Allowing real thresholds/weights instead of integer would not increase expressive power)

## (Knowledge-independent) tooth operators

### Definition (Values of individuals in knowledge-independent tooth operators)

Let  $\mathbf{C} = \mathbb{W}^t((C_i : w_i)_{i=1\dots m})$  be a **knowledge-independent** tooth operator, let  $I$  be an interpretation, and let  $d \in \Delta^I$  be an **individual** in the domain of  $I$ . Then

$$v_{\mathbf{C}}^I(d) = \sum_{i \in \{1, \dots, m\}} \{w_i \mid d \in C_i^I\}.$$

### Definition (Semantics of knowledge-independent tooth operators)

Let  $\mathbf{C} = \mathbb{W}^t((C_i : w_i)_{i=1\dots n})$  be a **knowledge-independent** tooth operator and let  $I$  be an interpretation. Then

$$\mathbf{C}^I = \{d \in \Delta^I \mid v_{\mathbf{C}}^I(d) \geq t\}.$$

## Entailment of (knowledge-independent) tooth expressions

### Definition (Evaluating knowledge-independent tooth operators over knowledge bases)

Let  $\mathcal{K}$  be a knowledge base, let  $a$  be an **individual name**, and let  $\mathbf{C} = \nabla^t((C_i : w_i)_{i=1\dots n})$  be a knowledge-independent tooth expression. Then we define the value of  $a$  in  $\mathbf{C}$  with respect to  $\mathcal{K}$  as

$$\mu_{\mathbf{C}}^{\mathcal{K}}(a) = \min\{v_{\mathbf{C}}^I(a^I) : I \models \mathcal{K}\}.$$

### Proposition

Let  $\mathcal{K}$  be any knowledge base, let  $a$  be an **individual name**, and let  $\mathbf{C} = \nabla^t((C_i : w_i)_{i=1\dots n})$  be any knowledge-independent tooth expression. Then

$$\mathcal{K} \models \mathbf{C}(a) \Leftrightarrow \mu_{\mathbf{C}}^{\mathcal{K}}(a) \geq t.$$

## Example

- Course A: 1 credit
- Course B: 1 credit
- Course C: 2 credits
- Course D: 2 credits
- A student must gain at least 3 credits.
- **Alice** took courses C and D
- **Bob** is a student who did not take course C.

$$K = \{\mathbf{Student} \sqsubseteq \forall^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2), \mathbf{Alice} : \mathbf{C}, \mathbf{Alice} : \mathbf{D}, \mathbf{Bob} : \mathbf{Student}, \mathbf{Bob} : \neg\mathbf{C}\}$$

Does it follow that...

- 1 Alice is a student?
- 2 Bob took course A?
- 3 Bob took course D?

## Example

- Course A: 1 credit
- Course B: 1 credit
- Course C: 2 credits
- Course D: 2 credits
- A student must gain at least 3 credits.
- **Alice** took courses C and D
- **Bob** is a student who did not take course C.

$$K = \{\mathbf{Student} \sqsubseteq \forall^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2), \mathbf{Alice} : \mathbf{C}, \mathbf{Alice} : \mathbf{D}, \mathbf{Bob} : \mathbf{Student}, \mathbf{Bob} : \neg\mathbf{C}\}$$

Does it follow that...

- 1 Alice is a student?
- 2 Bob took course A?
- 3 Bob took course D?



## Example

$$K = \{\mathbf{Student} \sqsubseteq \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2), \mathbf{Alice} : \mathbf{C}, \mathbf{Alice} : \mathbf{D}, \mathbf{Bob} : \mathbf{Student}, \mathbf{Bob} : \neg\mathbf{C}\}$$

$K \not\models \mathbf{Alice} : \mathbf{Student}$ .

Indeed, let  $I$  be any interpretation such that  $I \models K$ . Then  $\mathbf{Alice}^I \in \mathbf{C}^I$  and  $\mathbf{Alice}^I \in \mathbf{D}^I$ , and so

$$v_{(\mathbf{A}:1, \mathbf{B}:1, \mathbf{C}:2, \mathbf{D}:2)}^I(\mathbf{Alice}^I) = \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{Alice}^I \in \mathbf{A}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{Alice}^I \in \mathbf{B}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 2 & \text{if } \mathbf{Alice}^I \in \mathbf{C}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 2 & \text{if } \mathbf{Alice}^I \in \mathbf{D}^I \\ 0 & \text{otherwise} \end{array} \right\} \geq 4$$

and therefore  $\mathbf{Alice}^I \in (\mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2))^I = \{d \in \Delta^I : v_{(\mathbf{A}:1, \mathbf{B}:1, \mathbf{C}:2, \mathbf{D}:2)}^I(d) \geq 3\}$ .

But even though  $\mathbf{Student}^I \subseteq (\mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2))^I$  and  $\mathbf{Alice}^I \in (\mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2))^I$ , it does not follow that Alice is a student!

## Example

$$K = \{\mathbf{Student} \sqsubseteq \nabla^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2), \text{Alice} : \mathbf{C}, \text{Alice} : \mathbf{D}, \text{Bob} : \mathbf{Student}, \text{Bob} : \neg\mathbf{C}\}$$

$K \not\models \text{Bob} : \mathbf{A}$  either.

To be a student, Bob needs only 3 credits, and he can get them by taking courses **B** and **D** instead.

Let  $I$  be such that

- $\Delta^I = \{\text{alice}, \text{bob}\}$ ,  $\text{Alice}^I = \text{alice}$ ,  $\text{Bob}^I = \text{bob}$ ;
- $\mathbf{Student}^I = \{\text{bob}\}$ ;
- $\mathbf{A}^I = \emptyset$ ;
- $\mathbf{B}^I = \{\text{bob}\}$ ;
- $\mathbf{C}^I = \{\text{alice}\}$ ;
- $\mathbf{D}^I = \{\text{alice}, \text{bob}\}$ .

Then  $\nabla^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2))^I = \{\text{alice}, \text{bob}\} \supseteq \mathbf{Student}^I = \{\text{Bob}^I\}$ ;  $\text{Alice}^I \in C^I \cap D^I$ ;  
 $\text{Bob}^I \in \mathbf{Student}^I \setminus C^I$ .

## Example

$$K = \{\mathbf{Student} \sqsubseteq \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2), \mathbf{Alice} : \mathbf{C}, \mathbf{Alice} : \mathbf{D}, \mathbf{Bob} : \mathbf{Student}, \mathbf{Bob} : \neg\mathbf{C}\}$$

$K \models \mathbf{Bob} : \mathbf{D}$ .

Indeed, suppose that  $I \models K$ . Then  $\mathbf{Bob}^I \in \mathbf{Student}^I$ , so  $\mathbf{I} \in \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2)^I$  and

$$v_{(\mathbf{A}:1, \mathbf{B}:1, \mathbf{C}:2, \mathbf{D}:2)}^I(\mathbf{Bob}^I) = \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{Bob}^I \in \mathbf{A}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{Bob}^I \in \mathbf{B}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 2 & \text{if } \mathbf{Bob}^I \in \mathbf{C}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 2 & \text{if } \mathbf{Bob}^I \in \mathbf{D}^I \\ 0 & \text{otherwise} \end{array} \right\} \geq 3$$

which is only possible if  $\mathbf{Bob}^I \in \mathbf{D}^I$ .

## Example

$$K = \{\mathbf{Student} \sqsubseteq \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2), \mathbf{Alice} : \mathbf{C}, \mathbf{Alice} : \mathbf{D}, \mathbf{Bob} : \mathbf{Student}, \mathbf{Bob} : \neg\mathbf{C}\}$$

$K \models \mathbf{Bob} : \mathbf{D}$ .

Indeed, suppose that  $I \models K$ . Then  $\mathbf{Bob}^I \in \mathbf{Student}^I$ , so  $\mathbf{I} \in \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2)^I$  and

$$v_{(\mathbf{A}:1, \mathbf{B}:1, \mathbf{C}:2, \mathbf{D}:2)}^I(\mathbf{Bob}^I) = \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{Bob}^I \in \mathbf{A}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{Bob}^I \in \mathbf{B}^I \\ 0 & \text{otherwise} \end{array} \right\} + \left\{ \begin{array}{ll} 2 & \text{if } \mathbf{Bob}^I \in \mathbf{D}^I \\ 0 & \text{otherwise} \end{array} \right\} \geq 3$$

which is only possible if  $\mathbf{Bob}^I \in \mathbf{D}^I$ .

## Negative Weights/Thresholds are Unnecessary

### Observation

$$\nabla^t(C_1 : w_1, \dots) \equiv \nabla^{t-w_1}(\neg C_1 : -w_1, \dots)$$

- 1 If  $d \in C_1^I$  then  $v_{(C_1:w_1,\dots)}^I(d) = w_1 + \text{rest}$ ,  $v_{(\neg C_1:-w_1,\dots)}^I(d) = 0 + \text{rest}$  and so  $v_{(C_1:w_1,\dots)}^I(d) = v_{(\neg C_1:-w_1,\dots)}^I(d) + w_1$ ;
- 2 If  $d \notin C_1^I$  then  $v_{(C_1:w_1,\dots)}^I(d) = 0 + \text{rest}$ ,  $v_{(\neg C_1:-w_1,\dots)}^I(d) = -w_1 + \text{rest}$  and so  $v_{(C_1:w_1,\dots)}^I(d) = v_{(\neg C_1:-w_1,\dots)}^I(d) + w_1$ ;

So in either case,  $v_{(\neg C_1:-w_1,\dots)}^I(d) \geq t \Leftrightarrow v_{(C_1:w_1,\dots)}^I(d) \geq t - w_1$ .

Therefore, we can change all negative weights into positive ones by negating the corr. concepts, adjusting threshold. If at the end the threshold is  $\leq 0$ , the whole expression is just  $\top$ ; otherwise, we have an equivalent expression with positive weights and threshold.

## Example

Course A: 1 credit

Course B: 1 credit

Course C: 2 credits

Course D: 2 credits

Cheated: -2 credits

A student must gain at least 3 credits.

Course A: 1 credit

Course B: 1 credit

Course C: 2 credits

Course D: 2 credits

Did not cheat: 2 credits

A student must gain at least 5 credits.

---

Student  $\sqsubseteq \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2, \mathbf{Cheat} : -2)$

Student  $\sqsubseteq \mathbb{W}^5(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2, \neg\mathbf{Cheat} : 2)$

## Example

Course A: 1 credit

Course B: 1 credit

Course C: 2 credits

Course D: 2 credits

Cheated: -2 credits

A student must gain at least 3 credits.

Course A: 1 credit

Course B: 1 credit

Course C: 2 credits

Course D: 2 credits

Did not cheat: 2 credits

A student must gain at least 5 credits.

---

Student  $\sqsubseteq \mathbb{W}^3(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2, \mathbf{Cheat} : -2)$

Student  $\sqsubseteq \mathbb{W}^5(\mathbf{A} : 1, \mathbf{B} : 1, \mathbf{C} : 2, \mathbf{D} : 2, \neg\mathbf{Cheat} : 2)$

## Translating Threshold Expressions

### Our Theorem

Let  $\mathcal{L}$  be a Description Logic that contains all Boolean connectives, let  $\mathcal{K}$  be a  $\mathcal{L}(\mathbb{N})$  knowledge base and let  $\phi$  be a  $\mathcal{L}(\mathbb{N})$  axiom (positive integer weights). 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  $\phi_L$ .

**A NOTE:** In the translation, the size of a formula contains also the length of thresholds and weights (in binary).



# Translating Threshold Expressions

## The Main Lemma

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

Then we can build a knowledge base  $\mathcal{K}(T \mapsto \text{TOOTH})$  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 TOOTH) such that

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

### A Consequence of the Main Lemma

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  $\mathcal{K}_C$ .

Proof idea: Iteratively remove threshold expressions  $T$  in  $C$  from the “inside out”, adding new atomic symbols  $T\text{OOTH}$  and axioms  $\mathcal{K}(T \mapsto T\text{OOTH})$  to  $\mathcal{C}$  as needed.

## Translating Threshold Expressions

### Our Theorem

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 (integer weights). 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  $\phi_L$ .

Proof idea: translate every top-level concept  $C \in \mathcal{L}(\mathbb{W})$  appearing in  $\mathcal{K}$  into some  $C' \in \mathcal{L}$ , adding the required  $\mathcal{K}_C$  to  $\mathcal{K}$ , and then replace every  $C$  with the corresponding  $C'$  in the original  $\mathcal{K}$ .

## Proving the Main Lemma

### The Main Lemma

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

Then we can build a knowledge base  $\mathcal{K}(T \mapsto \text{TOOTH})$  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 TOOTH) such that

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

**Proof Idea:** Represent the binary encodings of weights that apply to some individual  $d$  and of threshold via new concept symbols, then sum and compare (ripple carry adder).

## Parameters

Let

$$k = \lceil \log_2(\max(t, w_1, \dots, w_n)) \rceil + 1$$

be the number of bits necessary to encode each of  $t$  and of the  $w_i$  in binary.

### Example

Let  $\mathcal{C} = \mathbb{W}^4(A_1 : 1, A_2 : 2, A_3 : 3)$ . Then  $n = 3$ , because we have three arguments, and  $k = 3$ , since we only need three bits to encode 1, 2, 3 and 4 in binary.

## 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 atomic symbols.

Then let  $\mathcal{K}_0$  be the TBox containing the axioms

- $W_{ij} \equiv C_i$  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  $W_{ij} \equiv \perp$  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  $T_j \equiv \perp$  for the others.

## Encoding the Weights

### Lemma

$|\mathcal{K}_0| = kn + k$ . 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,

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

Example:  $\mathcal{C} = \mathbb{W}^4(A_1 : 1, A_2 : 2, A_3 : 3)$

The binary encodings of 1, 2, 3 and 4 are 001, 010, 011 and 100. Therefore,  $\mathcal{K}_0$  contains the axioms  $W_{10} \equiv A_1$ ,  $W_{11} \equiv \perp$ ,  $W_{12} \equiv \perp$ ,  $W_{20} \equiv \perp$ ,  $W_{21} \equiv A_2$ ,  $W_{22} \equiv \perp$ ,  $W_{30} \equiv A_3$ ,  $W_{31} \equiv A_3$ ,  $W_{32} \equiv \perp$ ,  $T_0 \equiv \perp$ ,  $T_1 \equiv \perp$ ,  $T_2 \equiv \top$ .

The total number of axioms in  $\mathcal{K}_0$  is thus 12, that is,  $kn + k$  for  $k = n = 3$ .

## Encoding the Sum: Base Case

We define  $\mathcal{K}_1$  as the union of  $\mathcal{K}_0$  and the following axioms, for the fresh atomic symbols  $SUM_0^1 \dots SUM_{k-1}^1$ :

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

### Lemma

$|\mathcal{K}_1| = |\mathcal{K}_0| + k = kn + 2k$ . Let  $I$  be as before: then  $I$  has exactly one expansion to a model  $I'$  of  $\mathcal{K}_1$ , and for  $I'$  we have that

$$\sum \{2^j : j = 0 \dots k - 1, d \in (SUM_j^1)^{I'}\} = \sum \{w_i : 1 \leq i \leq 1, d \in C_i^I\}.$$

### Example (Continued)

$\mathcal{K}_1$  is  $\mathcal{K}_0$  plus the axioms  $SUM_0^1 \equiv W_{10}$ ,  $SUM_1^1 \equiv W_{11}$  and  $SUM_2^1 \equiv W_{12}$ . In total  $\mathcal{K}_1$  contains 15 axioms, as expected.



## Encoding the Sum: Other Cases

For  $i = 2 \dots n$ , we define inductively  $\mathcal{K}_i$  as  $\mathcal{K}_{i-1}$  plus the following axioms (for fresh symbols  $\text{SUM}_0^i \dots \text{SUM}_{k-1}^i$  and  $\text{CARRY}_0^i \dots \text{CARRY}_{k-1}^i$ ) and  $\text{OVERFLOW}^i$ :

- The axiom  $\text{CARRY}_0^i \equiv \perp$ ;
- For all  $j = 0 \dots k-1$ , the axiom

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

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

$$\text{CARRY}_j^i \equiv (\text{CARRY}_{j-1}^i \sqcap \text{SUM}_{j-1}^{i-1}) \sqcup (\text{CARRY}_{j-1}^i \sqcap W_{j-1}^i) \sqcup (\text{SUM}_{j-1}^{i-1} \sqcap W_{j-1}^i);$$

- The axiom

$$\text{OVERFLOW}^i \equiv (\text{CARRY}_{k-1}^i \sqcap \text{SUM}_{k-1}^{i-1}) \sqcup (\text{CARRY}_{k-1}^i \sqcap W_{k-1}^i) \sqcup (\text{SUM}_{k-1}^{i-1} \sqcap W_{k-1}^i).$$

**Note:** This constructs a *ripple-carry adder* via DL axioms.

## Encoding the Sum: Other Cases

### Lemma

For all  $\ell = 1 \dots n$ ,  $|\mathcal{K}_\ell| = kn + 2k + (\ell - 1)(2k + 1)$ , and so in particular

$$|\mathcal{K}_n| = kn + 2k + (n - 1)(2k + 1) = 3nk + n - 1.$$

Moreover, for every such  $\ell$ , every interpretation  $I$  as before can be extended in exactly one way to an interpretation  $I'$  which satisfies  $\mathcal{K}_\ell$ ; and for this interpretation  $SUM_{k-1}^\ell \dots SUM_0^\ell$  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  $d \in \Delta^{I'}$ )

$$\sum \{2^j : j = 0 \dots k - 1, d \in (SUM_j^\ell)^{I'}\} = \sum \{w_i : 1 \leq i \leq \ell, d \in C^I\}$$

whenever that value is less than  $2^k$ , and  $d \in (OVERFLOW^\ell)^{I'}$  otherwise

In particular, if  $I \models \mathcal{K}_n$  then

$$\sum \{2^j : j = 0 \dots k - 1, d \in (SUM_j^n)^{I'}\} = \sum \{w_i : 1 \leq i \leq n, d \in C^I\} = v_C^I(d)$$

is the value of our tooth expression  $C = \mathbb{W}^t(C_1 : w_1 \dots C_n : w_n)$  if that value is less than  $2^k$ , and otherwise  $d \in (OVERFLOW^n)^{I'}$  for at least one  $i = 2 \dots n$ .

## Encoding the Sum: Other Cases

### Example (Continued)

$\mathcal{K}_2$  adds to  $\mathcal{K}_1$  the axioms

$$16 \text{ CARRY}_0^2 \equiv \perp;$$

$$17 \text{ SUM}_0^2 \equiv (\text{CARRY}_0^2 \sqcap \text{SUM}_0^1 \sqcap W_0^2) \sqcup (\text{CARRY}_0^2 \sqcup \neg \text{SUM}_0^1 \sqcap \neg W_0^2) \sqcup (\neg \text{CARRY}_0^2 \sqcap \text{SUM}_0^1 \sqcap \neg W_0^2) \sqcup (\neg \text{CARRY}_0^2 \sqcap \neg \text{SUM}_0^1 \sqcap W_0^2);$$

$$18 \text{ SUM}_1^2 \equiv (\text{CARRY}_1^2 \sqcap \text{SUM}_1^1 \sqcap W_1^2) \sqcup (\text{CARRY}_1^2 \sqcup \neg \text{SUM}_1^1 \sqcap \neg W_1^2) \sqcup (\neg \text{CARRY}_1^2 \sqcap \text{SUM}_1^1 \sqcap \neg W_1^2) \sqcup (\neg \text{CARRY}_1^2 \sqcap \neg \text{SUM}_1^1 \sqcap W_1^2);$$

$$19 \text{ SUM}_2^2 \equiv (\text{CARRY}_2^2 \sqcap \text{SUM}_2^1 \sqcap W_2^2) \sqcup (\text{CARRY}_2^2 \sqcup \neg \text{SUM}_2^1 \sqcap \neg W_2^2) \sqcup (\neg \text{CARRY}_2^2 \sqcap \text{SUM}_2^1 \sqcap \neg W_2^2) \sqcup (\neg \text{CARRY}_2^2 \sqcap \neg \text{SUM}_2^1 \sqcap W_2^2);$$

$$20 \text{ CARRY}_1^2 \equiv (\text{CARRY}_0^2 \sqcap \text{SUM}_0^1) \sqcup (\text{CARRY}_0^2 \sqcap W_0^2) \sqcup (\text{SUM}_0^1 \sqcap W_0^2);$$

$$21 \text{ CARRY}_2^2 \equiv (\text{CARRY}_1^2 \sqcap \text{SUM}_1^1) \sqcup (\text{CARRY}_1^2 \sqcap W_1^2) \sqcup (\text{SUM}_1^1 \sqcap W_1^2);$$

$$22 \text{ OVERFLOW}^2 \equiv (\text{CARRY}_2^2 \sqcap \text{SUM}_2^1) \sqcup (\text{CARRY}_2^2 \sqcap W_2^2) \sqcup (\text{SUM}_2^1 \sqcap W_2^2).$$

## Encoding the Sum: Other Cases

### Example (Continued)

Then  $\mathcal{K}_3$  adds to this the axioms

$$\mathbf{23} \quad \text{CARRY}_0^3 \equiv \perp;$$

$$\mathbf{24} \quad \text{SUM}_0^3 \equiv (\text{CARRY}_0^3 \sqcap \text{SUM}_0^2 \sqcap W_0^3) \sqcup (\text{CARRY}_0^3 \sqcup \neg \text{SUM}_0^2 \sqcap \neg W_0^3) \sqcup (\neg \text{CARRY}_0^3 \sqcap \text{SUM}_0^2 \sqcap \neg W_0^3) \sqcup (\neg \text{CARRY}_0^3 \sqcap \neg \text{SUM}_0^2 \sqcap W_0^3);$$

$$\mathbf{25} \quad \text{SUM}_1^3 \equiv (\text{CARRY}_1^3 \sqcap \text{SUM}_1^2 \sqcap W_1^3) \sqcup (\text{CARRY}_1^3 \sqcup \neg \text{SUM}_1^2 \sqcap \neg W_1^3) \sqcup (\neg \text{CARRY}_1^3 \sqcap \text{SUM}_1^2 \sqcap \neg W_1^3) \sqcup (\neg \text{CARRY}_1^3 \sqcap \neg \text{SUM}_1^2 \sqcap W_1^3);$$

$$\mathbf{26} \quad \text{SUM}_2^3 \equiv (\text{CARRY}_2^3 \sqcap \text{SUM}_2^2 \sqcap W_2^3) \sqcup (\text{CARRY}_2^3 \sqcup \neg \text{SUM}_2^2 \sqcap \neg W_2^3) \sqcup (\neg \text{CARRY}_2^3 \sqcap \text{SUM}_2^2 \sqcap \neg W_2^3) \sqcup (\neg \text{CARRY}_2^3 \sqcap \neg \text{SUM}_2^2 \sqcap W_2^3);$$

$$\mathbf{27} \quad \text{CARRY}_1^3 \equiv (\text{CARRY}_0^3 \sqcap \text{SUM}_0^2) \sqcup (\text{CARRY}_0^3 \sqcap W_0^3) \sqcup (\text{SUM}_0^2 \sqcap W_0^3);$$

$$\mathbf{28} \quad \text{CARRY}_2^3 \equiv (\text{CARRY}_1^3 \sqcap \text{SUM}_1^2) \sqcup (\text{CARRY}_1^3 \sqcap W_1^3) \sqcup (\text{SUM}_1^2 \sqcap W_1^3);$$

$$\mathbf{29} \quad \text{OVERFLOW}^3 \equiv (\text{CARRY}_2^3 \sqcap \text{SUM}_2^2) \sqcup (\text{CARRY}_2^3 \sqcap W_2^3) \sqcup (\text{SUM}_2^2 \sqcap W_2^3).$$

## Comparing with the Threshold

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

- $\text{EQ}_{k-1} \equiv ((\text{SUM}_{k-1}^n \sqcap T_{k-1}) \sqcup (\neg \text{SUM}_{k-1}^n \sqcap \neg T_{k-1}));$

- For  $j = (k-2) \dots 0$ , the axiom

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

- $\text{MAJ}_{k-1} \equiv \text{SUM}_{k-1}^n \sqcap \neg T_{k-1};$

- For  $j = (k-2) \dots 0$ , the axiom

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

- The axiom

$$\text{TOOTH} \equiv \text{OVERFLOW}^2 \sqcup \dots \sqcup \text{OVERFLOW}^n \sqcup \text{MAJ}_{k-1} \sqcup \dots \sqcup \text{MAJ}_0 \sqcup \text{EQ}_0.$$

## Comparing with the Threshold

### Lemma

$|\mathcal{K}| = |\mathcal{K}_n| + 2k + 1 = 3nk + n + 2k$ . Moreover, every interpretation  $I$  as before 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  $j = k - 1 \dots 0$ ,  $d \in EQ_j^I$  if and only if the binary encodings of  $v_C^I(d)$  and of  $t$  agree from the most significant digit to the  $j$ -th least significant digit;
- For all  $j = k - 1 \dots 0$ ,  $d \in MAJ_j^I$  if and only if the binary encodings of  $v_C^I(d)$  and of  $t$  disagree on the  $j$ -th least significant digit, which is greater for  $v_C^I(d)$  than for  $t$ , but agree on all the digits on the left of it;
- $d \in TOOTH^{I'}$  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  $v_C^I(d)$  is greater than the threshold), or if there is a digit that is greater for  $v_C^I(d)$  than for  $t$  and all the digits to the left agree, or if all the digits of  $v_C^I(d)$  and of  $t$  are the same - that is, if and only if  $v_C^I(d) \geq t$ .

### Example (Continued)

We obtain  $\mathcal{K}$  by adding to  $\mathcal{K}_n$  the axioms

$$30 \quad \text{EQ}_2 \equiv ((\text{SUM}_2^3 \cap T_2) \sqcup (\neg \text{SUM}_2^3 \cap \neg T_2));$$

$$31 \quad \text{EQ}_1 \equiv \text{EQ}_2 \cap ((\text{SUM}_1^3 \cap T_1) \sqcup (\neg \text{SUM}_1^3 \cap \neg T_1));$$

$$32 \quad \text{EQ}_0 \equiv \text{EQ}_1 \cap ((\text{SUM}_0^3 \cap T_0) \sqcup (\neg \text{SUM}_0^3 \cap \neg T_0));$$

$$33 \quad \text{MAJ}_2 \equiv \text{SUM}_2^3 \cap \neg T_2;$$

$$34 \quad \text{MAJ}_1 \equiv \text{EQ}_2 \cap \text{SUM}_1^3 \cap \neg T_1;$$

$$35 \quad \text{MAJ}_0 \equiv \text{EQ}_1 \cap \text{SUM}_0^3 \cap \neg T_0;$$

$$36 \quad \text{TOOTH} \equiv \text{OVERFLOW}^2 \sqcup \text{OVERFLOW}^3 \sqcup \text{MAJ}_2 \sqcup \text{MAJ}_1 \sqcup \text{MAJ}_0 \sqcup \text{EQ}_0.$$

## Threshold Expressions are Cheap. Are they Good?

### What We Showed

If  $\mathcal{L}$  is a Description Logic containing Boolean connectives, the inference problem for  $\mathcal{L}(\mathbb{W})$  is no harder than the inference problem for  $\mathcal{L}$ .

### What We Implemented

We implemented the above-described polynomial translation algorithm for OWL. Works fine.

### What Remains to be Shown

Are simple threshold expression powerful enough to represent useful concepts? Can they be learned from data?



## Threshold Expressions are Cheap. Are they Good?

### What We Showed

If  $\mathcal{L}$  is a Description Logic containing Boolean connectives, the inference problem for  $\mathcal{L}(\mathbb{W})$  is no harder than the inference problem for  $\mathcal{L}$ .

### What We Implemented

We implemented the above-described polynomial translation algorithm for OWL. Works fine.

### What Remains to be Shown

Are simple threshold expression powerful enough to represent useful concepts? Can they be learned from data?

## The Gene Ontology

- Widely used ontology for annotating genes and gene products;
- More than 44,000 terms, split in three sub-ontologies
  - ▶ **Molecular Function:** **WHAT?** – Molecular-level activities, e.g. Transporter activity.
  - ▶ **Cellular Component:** **WHERE?** – Elements inside a cell, e.g. Mitochondrion
  - ▶ **Biological Process:** **WHY?** – Biological “purpose”, e.g. DNA repair.
- Various **mapping files** relating gene/gene products to Gene Ontology terms.
- Various “slim” sub-ontologies containing terms related to specific species or topics (e.g. Yeast subset, Plant subset. Metagenomics subset...). For example, according to the Saccharomyces Genome Database the enzyme ATP8 (ATP Synthase):
  - ▶ Executes molecular functions GO:0016887 ATPase activity, GO:0016787 Hydrolase activity, ...
  - ▶ Is located in GO:0005740 Mitochondrial Envelope, in GO:0005739 Mitochondrion, in GO:0005737 Cytoplasm, ...
  - ▶ Is involved in the processes of GO:0006811 Ion transport, GO:0055085 Transmembrane transport, ...

For our experiment, we focused on yeast sub-ontology and yeast-related mappings and considered this (difficult) problem:

To which degree can we infer Molecular Function annotations (WHAT?) from Cellular Component (WHERE?) and Biological Process (WHY?) ones?

## The Gene Ontology

- Widely used ontology for annotating genes and gene products;
- More than 44,000 terms, split in three sub-ontologies
  - ▶ **Molecular Function:** **WHAT?** – Molecular-level activities, e.g. Transporter activity.
  - ▶ **Cellular Component:** **WHERE?** – Elements inside a cell, e.g. Mitochondrion
  - ▶ **Biological Process:** **WHY?** – Biological “purpose”, e.g. DNA repair.
- Various **mapping files** relating gene/gene products to Gene Ontology terms.
- Various “slim” sub-ontologies containing terms related to specific species or topics (e.g. Yeast subset, Plant subset. Metagenomics subset...). For example, according to the Saccharomyces Genome Database the enzyme ATP8 (ATP Synthase):
  - ▶ Executes molecular functions GO:0016887 ATPase activity, GO:0016787 Hydrolase activity, ...
  - ▶ Is located in GO:0005740 Mitochondrial Envelope, in GO:0005739 Mitochondrion, in GO:0005737 Cytoplasm, ...
  - ▶ Is involved in the processes of GO:0006811 Ion transport, GO:0055085 Transmembrane transport, ...

For our experiment, we focused on yeast sub-ontology and yeast-related mappings and considered this (difficult) problem:

To which degree can we infer Molecular Function annotations (WHAT?) from Cellular Component (WHERE?) and Biological Process (WHY?) ones?

## The Gene Ontology

- Widely used ontology for annotating genes and gene products;
- More than 44,000 terms, split in three sub-ontologies
  - ▶ **Molecular Function:** **WHAT?** – Molecular-level activities, e.g. Transporter activity.
  - ▶ **Cellular Component:** **WHERE?** – Elements inside a cell, e.g. Mitochondrion
  - ▶ **Biological Process:** **WHY?** – Biological “purpose”, e.g. DNA repair.
- Various **mapping files** relating gene/gene products to Gene Ontology terms.
- Various “slim” sub-ontologies containing terms related to specific species or topics (e.g. Yeast subset, Plant subset. Metagenomics subset...). For example, according to the Saccharomyces Genome Database the enzyme ATP8 (ATP Synthase):
  - ▶ Executes molecular functions GO:0016887 ATPase activity, GO:0016787 Hydrolase activity, ...
  - ▶ Is located in GO:0005740 Mitochondrial Envelope, in GO:0005739 Mitochondrion, in GO:0005737 Cytoplasm, ...
  - ▶ Is involved in the processes of GO:0006811 Ion transport, GO:0055085 Transmembrane transport, ...

For our experiment, we focused on yeast sub-ontology and yeast-related mappings and considered this (difficult) problem:

To which degree can we infer Molecular Function annotations (WHAT?) from Cellular Component (WHERE?) and Biological Process (WHY?) ones?

## The purpose of the experiment

To which degree can we infer Molecular Function annotations (WHAT?) from Cellular Component (WHERE?) and Biological Process (WHY?) ones?

- This is a **hard problem**: generally, you'd use much more information than Cellular Component and Biological Process annotations to try to guess the function of a protein.
- Nonetheless, there are clear correlations between the annotations of the three sub-ontologies, ones that “classic” machine learning approaches can easily find.
- **Our purpose**: We want to show that simple ( $\leq 10$  components) threshold operators, learned via a very basic evolutionary algorithm, can capture such correlations about as well as more sophisticated approaches.

**NOTE**: we are **not** trying to come up with a competitive, novel ML algorithm. We are exploring the expressive power of simple threshold expressions in real life applications. If our system will perform about as well as the competitors, we will be happy.

## The purpose of the experiment

To which degree can we infer Molecular Function annotations (WHAT?) from Cellular Component (WHERE?) and Biological Process (WHY?) ones?

- This is a **hard problem**: generally, you'd use much more information than Cellular Component and Biological Process annotations to try to guess the function of a protein.
- Nonetheless, there are clear correlations between the annotations of the three sub-ontologies, ones that “classic” machine learning approaches can easily find.
- **Our purpose**: We want to show that simple ( $\leq 10$  components) threshold operators, learned via a very basic evolutionary algorithm, can capture such correlations about as well as more sophisticated approaches.

**NOTE**: we are **not** trying to come up with a competitive, novel ML algorithm. We are exploring the expressive power of simple threshold expressions in real life applications. If our system will perform about as well as the competitors, we will be happy.

## Data Preparation and Evaluation

- Remove annotations marked as “dubious”;
- Remove top-level annotations (“Protein ATP8 is in **some** cellular components”);
- Remove products with  $< 3$  Cellular Component or Biological Process annotations (“features”);
- Consider Molecular Function annotations (“labels”) that apply to at least 100 products.

**What remains:** 4,595 gene products, 120 features, 17 possible labels.

- We kept five random labels for final testing, used the others to implement our approach and tuning the baseline algorithms.

This dataset is **heavily imbalanced**: most molecular functions apply to few proteins. This requires some adjustments to some baseline methods (e.g. oversampling positive cases) for good performance.

As performance metric, we use the Matthews Correlation Coefficient (Pearson Correlation on binary classification problems), which is more appropriate for imbalanced classes than e.g. accuracy or  $F_1$ .

## Data Preparation and Evaluation

- Remove annotations marked as “dubious”;
- Remove top-level annotations (“Protein ATP8 is in **some** cellular components”);
- Remove products with  $< 3$  Cellular Component or Biological Process annotations (“features”);
- Consider Molecular Function annotations (“labels”) that apply to at least 100 products.

**What remains:** 4,595 gene products, 120 features, 17 possible labels.

- We kept five random labels for final testing, used the others to implement our approach and tuning the baseline algorithms.

This dataset is **heavily imbalanced**: most molecular functions apply to few proteins. This requires some adjustments to some baseline methods (e.g. oversampling positive cases) for good performance.

As performance metric, we use the Matthews Correlation Coefficient (Pearson Correlation on binary classification problems), which is more appropriate for imbalanced classes than e.g. accuracy or  $F_1$ .



## Data Preparation and Evaluation

- Remove annotations marked as “dubious”;
- Remove top-level annotations (“Protein ATP8 is in **some** cellular components”);
- Remove products with  $< 3$  Cellular Component or Biological Process annotations (“features”);
- Consider Molecular Function annotations (“labels”) that apply to at least 100 products.

**What remains:** 4,595 gene products, 120 features, 17 possible labels.

- We kept five random labels for final testing, used the others to implement our approach and tuning the baseline algorithms.

This dataset is **heavily imbalanced**: most molecular functions apply to few proteins. This requires some adjustments to some baseline methods (e.g. oversampling positive cases) for good performance.

As performance metric, we use the Matthews Correlation Coefficient (Pearson Correlation on binary classification problems), which is more appropriate for imbalanced classes than e.g. accuracy or  $F_1$ .

## Algorithms, Training, Results

We used the following classifiers as implemented in the Waikato Environment for Knowledge Analysis (WEKA):

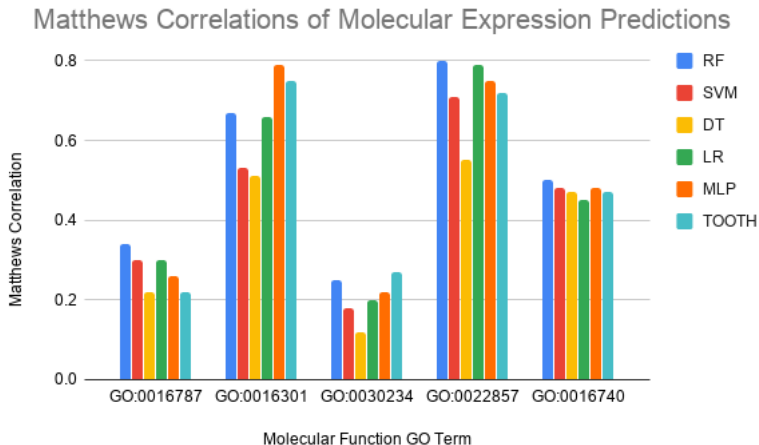
- Random Forests (RM);
- Support Vector Machine (SVM);
- Decision table majority classifier (DT);
- Logistic regression classifier (LR);
- Multilayer perceptron classifier (MLP).

Plus, of course,

- Our simple threshold expression learning algorithm (TOOTH).

For all labels, we split all our data in five folds, maintaining the same proportion of true labels in all folds, trained all algorithms on four and tested the other, averaged; and we used the 12 non-final labels to tune the parameters of the algorithms for this kind of problem, reserving the five last for evaluation.

## Results (average over five folds)



**NOTE:** LR is a linear classifier, so we could translate its models into threshold expressions.

## Conclusions

Threshold Expressions may be “syntactic sugar”, but they are **really sweet!**

- Arise naturally in knowledge representation tasks;
- Can represent compactly and readably complex concepts;
- Do not make the inference problem any harder (for most DLs);
- Can be learned from data in real life examples.
- Simple, effective link between Machine Learning and Symbolic Reasoning.

What's next:

- Better integration with OWLAPI/Protege
- Richer threshold expressions: thresholds on relations, e.g. +3 points for each minor child, +1 point for each adult child;
- Explore and compare with decision trees (human interpretability).