PRACTICAL USES OF EXISTENTIAL RULES IN KNOWLEDGE REPRESENTATION

Part 2: Solving Horn-$\mathcal{ALC}$ Classification with Existential Rules

David Carral,[1] Markus Krötzsch,[1] and Jacopo Urbani[2]
1. TU Dresden
2. VU University Amsterdam

Special thanks to Irina Dragoste,[1] Ceriel Jacobs,[2] and Maximilian Marx[1]
for their invaluable contributions to the software used in this tutorial

KR 2020, September 13, 2020

Outline

Goal

Show some example where either **rules** or **related ideas** were crucial to achieve the state of the art

- Horn-$\mathcal{ALC}$ reasoning
- PLP
- Data integration
- Stream reasoning

1^{st} Scenario: Horn-$\mathcal{ALC}$ reasoning

The Description Logic Horn-$\mathcal{ALC}$: Syntax

Definition. A Horn-$\mathcal{ALC}$ ontology is a set of Horn-$\mathcal{ALC}$ axioms:

$$A \sqsubseteq \bot \qquad \top \sqsubseteq B \qquad A \sqsubseteq B \qquad A \sqcap E \sqsubseteq B \qquad \exists R.A \sqsubseteq B \qquad A \sqsubseteq \forall R.B \qquad A \sqsubseteq \exists R.B$$

In the above; A, B, and E are concept names; and R is a role name.

Remark. Note the axioms of the form $A \sqsubseteq \forall R.B$, which are not $\mathcal{EL}$, such as:

$$\mathsf{CheesePizza} \sqsubseteq \forall \mathsf{HasTopping}.\mathsf{Cheese}$$

The axiom states that “all toppings in a cheese pizza are cheese toppings”.

Even though Horn-$\mathcal{ALC}$ is not much more expressive than $\mathcal{EL}$, (Krötzsch, Rudolph, and Hitzler 2013) have showed that:

Theorem. Solving classification over Horn-$\mathcal{ALC}$ is ExpTime-complete.

A Consequence-Based Calculus to Solve Classification

$$R^C_A)\ \frac{}{A \sqsubseteq A} : A \in \text{Concepts}(\mathcal{O}) \qquad R^+_\exists)\ \frac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.B} : A \sqsubseteq \exists R.B \in \mathcal{O}$$

$$R^\exists_A)\ \frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad D \in \mathbb{D}}{\mathbb{D} \sqsubseteq D} \qquad R^-_\exists)\ \frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \exists R.A \sqsubseteq B \in \mathcal{O}$$

$$R^1_\sqcap)\ \frac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \top \sqsubseteq B \in \mathcal{O} \qquad R^\perp_\exists)\ \frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq \perp}{\mathbb{C} \sqsubseteq \perp}$$

$$R^1_\sqcap)\ \frac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : A \sqsubseteq B \in \mathcal{O} \qquad R_\forall)\ \frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.(\mathbb{D} \sqcap B)} : A \sqsubseteq \forall R.B \in \mathcal{O}$$

$$R^2_\sqcap)\ \frac{\mathbb{C} \sqsubseteq A \quad \mathbb{C} \sqsubseteq E}{\mathbb{C} \sqsubseteq B} : A \sqcap E \sqsubseteq B \in \mathcal{O}$$

Figure: Classification Calculus for Horn-$\mathcal{ALC}$. Where A, B, and E are concept names; R is a role name; and $\mathbb{C}$ and $\mathbb{D}$ are conjunctions of concept names

Remark. The above procedure is based on the work of (Kazakov 2009).

Consequence-Based Calculus: Complexity

Theorem. The Horn-$\mathcal{ALC}$ classification calculus runs in exponential time in the size of the input ontology $\mathcal{O}$.

Remark. Note that this calculus produces inferences of the form

$$(1)\quad \mathbb{C} \sqsubseteq B \qquad \text{and} \qquad (2)\quad \mathbb{C} \sqsubseteq \exists R.\mathbb{D}$$

where B is a concept name, R is a role name, and $\mathbb{C}$ and $\mathbb{D}$ are conjunctions of concept names. Therefore, on input $\mathcal{O}$, the calculus may produce at most

$$2^{|\text{Concepts}(\mathcal{O})|} \times |\text{Concepts}(\mathcal{O})| \quad \text{and} \quad 2 \times 2^{|\text{Concepts}(\mathcal{O})|} \times |\text{Roles}(\mathcal{O})|$$

inferences of type (1) and (2), respectively.

Remark. Since classification over Horn-$\mathcal{ALC}$ is an ExpTime-complete problem, the calculus is worst-case optimal.

Implementing the Classification Calculus: Datalog

Because of the following result, we can not implement the Horn-$\mathcal{ALC}$ classification calculus using a fixed Datalog rule set:

Theorem. The data complexity of fact entailment over Datalog is in P.

Assume that we can implement the Horn-$\mathcal{ALC}$ classification calculus with a fixed Datalog rule set (as we did for the $\mathcal{EL}$ classification calculus). Then:

1. By the above theorem, we could solve Horn-$\mathcal{ALC}$ classification in polynomial time.
2. By (1), we could solve an ExpTime-hard problem in polynomial time.
3. By (2), P = ExpTime ($\lightning$)

Remark. To implement the Horn-$\mathcal{ALC}$ classification calculus (or any other procedure that solves Horn-$\mathcal{ALC}$ classification), we need a rule-based language with ExpTime-hard data complexity!

Implementing the Classification Calculus: Datalog(S)

We study Datalog(S), an extension of Datalog that can model exponential computations.

Example. Consider the following Datalog(S) rule set:

$$\begin{aligned}
\text{Person}(x) &\rightarrow \text{LikesAll}(x, \emptyset) \\
\text{LikesAll}(x, S) \wedge \text{Likes}(x, y) &\rightarrow \text{LikesAll}(x, S \cup \{y\}) \\
\text{LikesAll}(x, S) &\rightarrow \text{AllLikeAll}(\{x\}, S) \\
\text{AllLikeAll}(S, T) \wedge \text{LikesAll}(x, T) &\rightarrow \text{AllLikeAll}(X \cup \{x\}, T) \\
\text{AllLikeAll}(S, S) \wedge \text{alice} \in S &\rightarrow \text{CliqueOfAlice}(S)
\end{aligned}$$

Theorem. Checking fact entailment for Datalog(S) is ExpTime-complete for both data and combined complexity.

See (Carral et al. 2019) for a complete proof of the above result.

Implementing the Classification Calculus: Datalog(S)

Using a function to encode the axioms and entities in an input ontology as facts and a fixed Datalog(S) rule set, we can implement the Horn-$\mathcal{ALC}$ classification calculus.

Example. For an ontology $\mathcal{O}$, let Facts($\mathcal{O}$) be the fact set such that:

$$A \sqsubseteq \bot \mapsto \texttt{nf:axiom}_{\sqsubseteq}(c_A, c_\bot) \qquad \exists R.A \sqsubseteq B \mapsto \texttt{nf:axiom}_{\exists\sqsubseteq}(c_A, c_R, c_B)$$

$$\top \sqsubseteq B \mapsto \texttt{nf:axiom}_{\sqsubseteq}(c_\top, c_B) \qquad A \sqsubseteq \forall R.B \mapsto \texttt{nf:axiom}_{\sqsubseteq\forall}(c_A, c_R, c_B)$$

$$A \sqsubseteq B \mapsto \texttt{nf:axiom}_{\sqsubseteq}(c_A, c_B) \qquad A \sqsubseteq \exists R.B \mapsto \texttt{nf:axiom}_{\sqsubseteq\exists}(c_A, c_R, c_B)$$

$$A \sqcap E \sqsubseteq B \mapsto \texttt{nf:axiom}_{\sqcap\sqsubseteq}(c_A, c_E, c_B) \qquad A \in \text{Concepts}(\mathcal{O}) \mapsto \texttt{nf:concept}(c_E)$$

In the above; c_A, c_B, c_E, $c_\top$, and $c_\bot$ are fresh constants unique for A, B, E, $\top$, and $\bot$, respectively; and c_R is a fresh constant unique R.

Implementing the Classification Calculus: Datalog(S)

We translate the production rules in the Horn-$\mathcal{ALC}$ classification calculus (left) into analogous Datalog(S) rules (right):

$\dfrac{}{A \sqsubseteq A} : A \in \text{Concepts}(\mathcal{O})$	$\texttt{nf:concept}(a) \rightarrow \texttt{SC}(\{a\}, a)$
$\dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D}}{\mathbb{D} \sqsubseteq D} : D \in \mathbb{D}$	$\texttt{Ex}(C, r, D) \wedge d \in D \rightarrow \texttt{SC}(D, d)$
$\dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \top \sqsubseteq B \in \mathcal{O}$	$\texttt{SC}(C, a) \wedge \texttt{nf:axiom}_{\sqsubseteq}(c_{\top}, b) \rightarrow \texttt{SC}(C, b)$
$\dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : A \sqsubseteq B \in \mathcal{O}$	$\texttt{SC}(C, a) \wedge \texttt{nf:axiom}_{\sqsubseteq}(a, b) \rightarrow \texttt{SC}(C, b)$
$\dfrac{\mathbb{C} \sqsubseteq A \quad \mathbb{C} \sqsubseteq E}{\mathbb{C} \sqsubseteq B} : A \sqcap E \sqsubseteq B \in \mathcal{O}$	$\texttt{SC}(C, a) \wedge \texttt{SC}(C, e) \wedge \texttt{nf:axiom}_{\sqcap\sqsubseteq}(a, e, b) \rightarrow \texttt{SC}(C, b)$

Implementing the Classification Calculus: Datalog(S)

We translate the production rules in the Horn-$\mathcal{ALC}$ classification calculus (left) into analogous Datalog(S) rules (right):

$$\frac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.B} : A \sqsubseteq \exists R.B \in \mathcal{O} \qquad \texttt{SC}(C, a) \wedge \texttt{nf:axiom}_{\sqsubseteq\exists}(a, r, b) \rightarrow \texttt{Ex}(C, r, \{b\})$$

$$\frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \exists R.A \sqsubseteq B \in \mathcal{O} \qquad \texttt{Ex}(C, r, D) \wedge \texttt{SC}(D, a) \wedge \texttt{nf:axiom}_{\exists\sqsubseteq}(r, a, b) \rightarrow \texttt{SC}(C, b)$$

$$\frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq \bot}{\mathbb{C} \sqsubseteq \bot} \qquad \texttt{Ex}(C, r, D) \wedge \texttt{SC}(D, c_\bot) \rightarrow \texttt{SC}(C, c_\bot)$$

$$\frac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.(\mathbb{D} \sqcap B)} : A \sqsubseteq \forall R.B \in \mathcal{O} \qquad \texttt{Ex}(C, r, D) \wedge \texttt{SC}(C, a) \wedge \texttt{nf:axiom}_{\sqsubseteq\forall}(a, r, b) \rightarrow \texttt{Ex}(C, r, D \cup \{b\})$$

Implementing the Classification Calculus: Datalog(S)

Definition. Let $\mathcal{R}_{\mathsf{HALC}}$ be the rule set containing all of the above rules:

$$\mathtt{nf{:}concept}(a) \rightarrow \mathtt{SC}(\{a\}, a) \qquad \mathtt{SC}(C, a) \wedge \mathtt{nf{:}axiom}_{\sqsubseteq}(c_\top, b) \rightarrow \mathtt{SC}(C, b)$$

$$\mathtt{Ex}(C, r, D) \wedge d \in D \rightarrow \mathtt{SC}(D, d) \qquad \mathtt{SC}(C, a) \wedge \mathtt{nf{:}axiom}_{\sqsubseteq}(a, b) \rightarrow \mathtt{SC}(C, b)$$

$$\mathtt{SC}(C, a) \wedge \mathtt{SC}(C, e) \wedge \mathtt{nf{:}axiom}_{\sqcap\sqsubseteq}(a, e, b) \rightarrow \mathtt{SC}(C, b)$$

$$\mathtt{SC}(C, a) \wedge \mathtt{nf{:}axiom}_{\sqsubseteq\exists}(a, r, b) \rightarrow \mathtt{Ex}(C, r, \{b\})$$

$$\mathtt{Ex}(C, r, D) \wedge \mathtt{SC}(D, a) \wedge \mathtt{nf{:}axiom}_{\exists\sqsubseteq}(r, a, b) \rightarrow \mathtt{SC}(C, b)$$

$$\mathtt{Ex}(C, r, D) \wedge \mathtt{SC}(D, c_\bot) \rightarrow \mathtt{SC}(C, c_\bot)$$

$$\mathtt{Ex}(C, r, D) \wedge \mathtt{SC}(C, a) \wedge \mathtt{nf{:}axiom}_{\sqsubseteq\forall}(a, r, b) \rightarrow \mathtt{Ex}(C, r, D \cup \{b\})$$

Theorem. Consider a Horn-$\mathcal{ALC}$ ontology $\mathcal{O}$ and an axiom of the form $A \sqsubseteq B$. Then, $\mathcal{O} \models A \sqsubseteq B$ if and only if $\mathcal{R}_{\mathsf{HALC}} \cup \mathsf{Facts}(\mathcal{O}) \models \mathtt{SC}(c_A, c_B)$.

Implementing the Classification Calculus: ∃-Rules

Alas, VLog does not support Datalog(S) reasoning. But there maybe some other rule-based language with ExpTime-hard data complexity that we can use!

The following result is a recent finding by (Krötzsch, Marx, and Rudolph 2019):

Theorem. The data complexity of fact entailment over rule sets that terminate with respect to the restricted chase is ExpTime-hard.

Remark. Note that the data complexity of fact entailment over existential rule sets that terminate with respect to the Skolem chase is in P.

(Carral et al. 2019) have proposed a translation from Datalog(S) into ∃-rules such that:

- The resulting rule sets terminate w.r.t. the **Datalog-first restricted chase.**
- Fact entailment is "preserved".

From Datalog(S) to Existential Rules

$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \tag{1.1}$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \tag{1.2}$$

From Datalog(S) to Existential Rules

$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset)$ $\quad$ $\mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \tag{1.1}$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \tag{1.2}$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \tag{2.1}$$

From Datalog(S) to Existential Rules

$$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$$

$$\rightarrow \exists V \,.\, empty(V) \tag{1.1}$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \tag{1.2}$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V \,.\, likesAll(x, V) \wedge SU(S, y, V) \tag{2.1}$$

$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

From Datalog(S) to Existential Rules

$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset) \qquad \mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \tag{1.1}$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \tag{1.2}$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \tag{2.1}$$

eve

$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

From Datalog(S) to Existential Rules

$$\text{Person}(x) \to \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \land \text{Likes}(x, y) \to \text{LikesAll}(x, S \cup \{y\})$$

$$\to \exists V \,.\, empty(V) \tag{1.1}$$

$$person(x) \land empty(Y) \to likesAll(x, Y) \tag{1.2}$$

$$likesAll(x, S) \land likes(x, y) \to \exists V \,.\, likesAll(x, V) \land SU(S, y, V) \tag{2.1}$$

eve

$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

$n_{\emptyset}$

From Datalog(S) to Existential Rules

$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . \mathit{empty}(V) \tag{1.1}$$

$$\mathit{person}(x) \wedge \mathit{empty}(Y) \rightarrow \mathit{likesAll}(x, Y) \tag{1.2}$$

$$\mathit{likesAll}(x, S) \wedge \mathit{likes}(x, y) \rightarrow \exists V . \mathit{likesAll}(x, V) \wedge \mathit{SU}(S, y, V) \tag{2.1}$$

$\mathit{person}(\mathit{eve})$
$\mathit{likes}(\mathit{eve}, a)$
$\mathit{likes}(\mathit{eve}, b)$

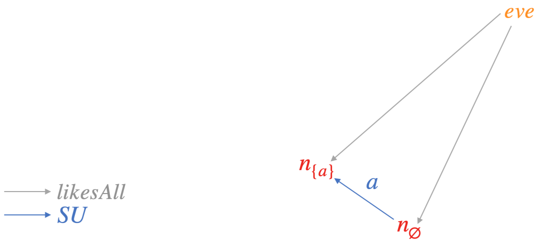
From Datalog(S) to Existential Rules

$$\text{Person}(x) \to \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \land \text{Likes}(x, y) \to \text{LikesAll}(x, S \cup \{y\})$$

$$\to \exists V . \mathit{empty}(V) \tag{1.1}$$

$$\mathit{person}(x) \land \mathit{empty}(Y) \to \mathit{likesAll}(x, Y) \tag{1.2}$$

$$\mathit{likesAll}(x, S) \land \mathit{likes}(x, y) \to \exists V . \mathit{likesAll}(x, V) \land \mathit{SU}(S, y, V) \tag{2.1}$$



$\mathit{person}(\mathit{eve})$
$\mathit{likes}(\mathit{eve}, a)$
$\mathit{likes}(\mathit{eve}, b)$

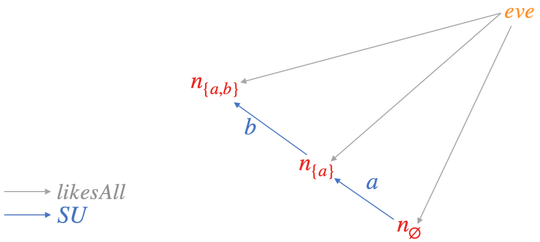
From Datalog(S) to Existential Rules

$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset) \qquad \mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \tag{1.1}$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \tag{1.2}$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \tag{2.1}$$



$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

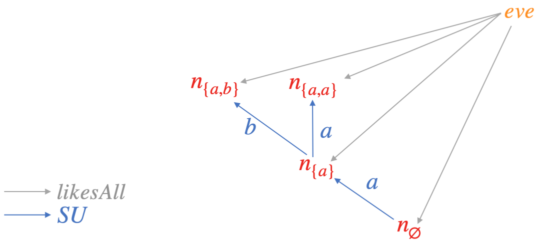
From Datalog(S) to Existential Rules

$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \quad (1.1)$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \quad (1.2)$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \quad (2.1)$$



$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

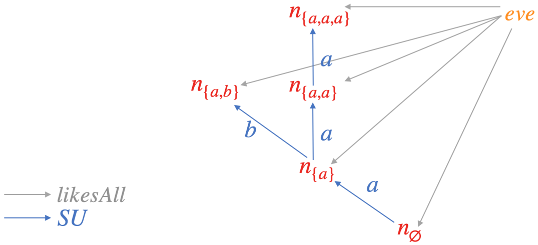
From Datalog(S) to Existential Rules

$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V\,.\, empty(V) \qquad (1.1)$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \qquad (1.2)$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V\,.\, likesAll(x, V) \wedge SU(S, y, V) \qquad (2.1)$$



$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

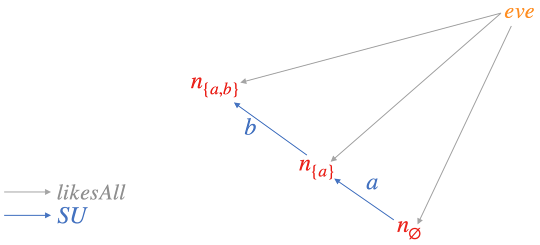
From Datalog(S) to Existential Rules

Person(x) → LikesAll$(x, \emptyset)$ LikesAll(x, S) ∧ Likes(x, y) → LikesAll$(x, S \cup \{y\})$

$$\rightarrow \exists V \,.\, empty(V) \quad (1.1)$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \quad (1.2)$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V \,.\, likesAll(x, V) \wedge SU(S, y, V) \quad (2.1)$$



$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

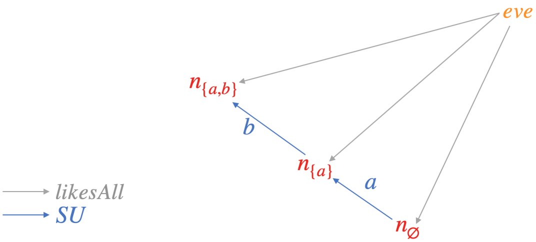
From Datalog(S) to Existential Rules

$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset) \qquad \mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V \,.\, empty(V) \tag{1.1}$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \tag{1.2}$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V \,.\, likesAll(x, V) \wedge SU(S, y, V) \wedge SU(V, y, V) \tag{2.1}$$



$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

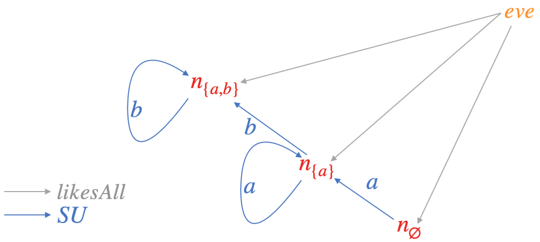
From Datalog(S) to Existential Rules

$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \quad (1.1)$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \quad (1.2)$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \wedge SU(V, y, V) \quad (2.1)$$



$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

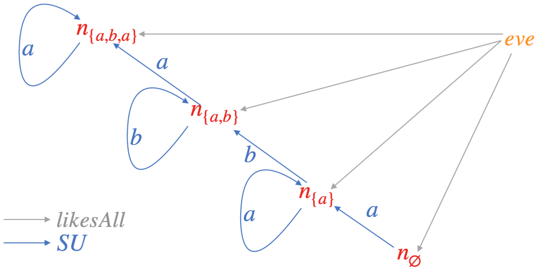
From Datalog(S) to Existential Rules

$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset)$ $\quad$ $\text{LikesAll}(x, S) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . \mathit{empty}(V) \quad (1.1)$$

$$\mathit{person}(x) \wedge \mathit{empty}(Y) \rightarrow \mathit{likesAll}(x, Y) \quad (1.2)$$

$$\mathit{likesAll}(x, S) \wedge \mathit{likes}(x, y) \rightarrow \exists V . \mathit{likesAll}(x, V) \wedge \mathit{SU}(S, y, V) \wedge \mathit{SU}(V, y, V) \quad (2.1)$$



person(eve)
likes(eve, a)
likes(eve, b)

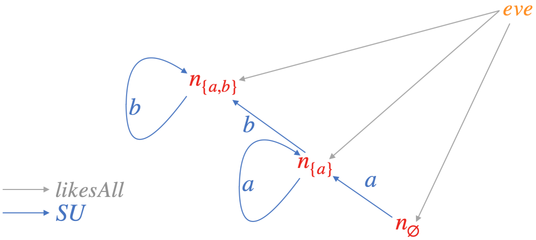
From Datalog(S) to Existential Rules

$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset) \qquad \mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \quad (1.1)$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \quad (1.2)$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \wedge SU(V, y, V) \quad (2.1)$$

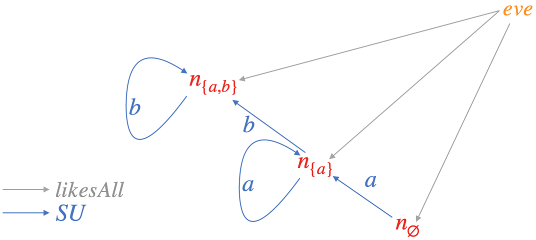


$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

From Datalog(S) to Existential Rules

$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset) \qquad \mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . \mathit{empty}(V) \tag{1.1}$$
$$\mathit{person}(x) \wedge \mathit{empty}(Y) \rightarrow \mathit{likesAll}(x, Y) \tag{1.2}$$
$$\mathit{likesAll}(x, S) \wedge \mathit{likes}(x, y) \rightarrow \exists V . \mathit{likesAll}(x, V) \wedge \mathit{SU}(S, y, V) \wedge \mathit{SU}(V, y, V) \tag{2.1}$$
$$\mathit{SU}(U, x, V) \wedge \mathit{SU}(U, y, U) \rightarrow \mathit{SU}(V, y, V) \tag{2.2}$$



$\mathit{person}(\mathit{eve})$
$\mathit{likes}(\mathit{eve}, a)$
$\mathit{likes}(\mathit{eve}, b)$

From Datalog(S) to Existential Rules

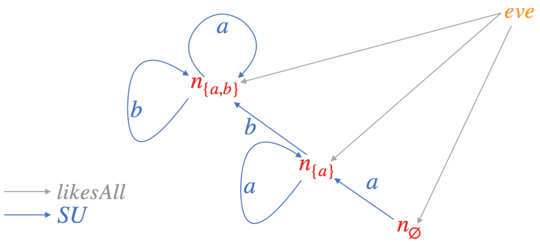
$\mathsf{Person}(x) \rightarrow \mathsf{LikesAll}(x, \emptyset) \qquad \mathsf{LikesAll}(x, S) \wedge \mathsf{Likes}(x, y) \rightarrow \mathsf{LikesAll}(x, S \cup \{y\})$

$$\rightarrow \exists V . empty(V) \quad (1.1)$$

$$person(x) \wedge empty(Y) \rightarrow likesAll(x, Y) \quad (1.2)$$

$$likesAll(x, S) \wedge likes(x, y) \rightarrow \exists V . likesAll(x, V) \wedge SU(S, y, V) \wedge SU(V, y, V) \quad (2.1)$$

$$SU(U, x, V) \wedge SU(U, y, U) \rightarrow SU(V, y, V) \quad (2.2)$$



$person(eve)$
$likes(eve, a)$
$likes(eve, b)$

Solving ExpTime-hard Problems

Step-by-step procedure to implement an ExpTime algorithm with VLog:

1. Encode input using a set of facts $\mathcal{F}$.
2. Encode ExpTime algorithm using a fixed Datalog(S) rule set $\mathcal{R}$.
3. Apply the translation by (Carral et al. 2019) to $\mathcal{R}$ to obtain a set $\mathcal{R}'$ of existential rules such that:
   - The rule set $\mathcal{R}'$ "preserves" fact entailment over $\mathcal{R}$.
   - The rule set $\mathcal{R}'$ terminates w.r.t. the Datalog-first restricted chase.
4. Use VLog to compute all of the consequences of $\mathcal{R}' \cup \mathcal{F}$.

Remark. For a detailed explanation of the above procedure, see (Carral, Dragoste, and Rudolph 2020).

Evaluation: Classification

ID	#Ax.	#SC	VLog	Konclude
00040	223K	1051K	432s	5s
00048	142K	718K	387s	3s
00477	318K	162K	1s	3s
00533	159K	965K	132s	2s
00786	152K	2283K	549s	14s

Figure: Ontologies and results for classification showing: axiom count, number of SC facts derived, and reasoning times for VLog and Konclude

Remark. Presented in (Carral et al. 2019).

Evaluation: Class Retrieval

Definition. A Horn-$\mathcal{ALC}$ ontology is a set of Horn-$\mathcal{ALC}$ axioms:

$$A \sqsubseteq \bot \qquad \top \sqsubseteq B \qquad A \sqsubseteq B \qquad A \sqcap E \sqsubseteq B$$

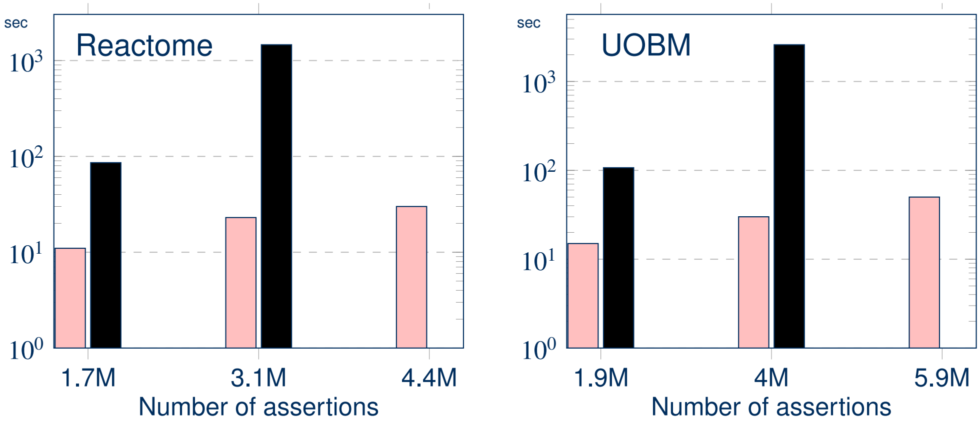
$$\exists R.A \sqsubseteq B \qquad A \sqsubseteq \forall R.B \qquad A \sqsubseteq \exists R.B \qquad A(a) \qquad R(a, b)$$

Where A, B, and E are concepts; R is a role; and a and b are individuals.

Definition. Class retrieval is the reasoning task of computing all axioms of the form $A(a)$ that are logically entailed by some input ontology $\mathcal{O}$.

Remark. The Horn-$\mathcal{ALC}$ classification calculus can be extended with 3 rules (as done by (Carral et al. 2019)) to solve class retrieval.

Evaluation: Class Retrieval



Experimental results for class retrieval for VLog (pink/grey) and Konclude (black)

Remark. Presented in (Carral et al. 2019).

Conclusions and Future Work

Remark. We can use VLog to solve (ExpTime-)hard problems!

Future work:

- Rulewerk Extension: translate Datalog(S) to existential rules
- VLog Extension: native support for Datalog(S)
- Implement existing calculi using our approach

References I

- Carral, David, Irina Dragoste, and Sebastian Rudolph (2020). “Reasoner = Logical Calculus + Rule Engine”. In: **Künstliche Intell.**
- Carral, David et al. (2019). “Chasing Sets: How to Use Existential Rules for Expressive Reasoning”. In: **P. of the 28th Int. Joint Conf. on AI (IJCAI 2019)**.
- Kazakov, Yevgeny (2009). “Consequence-Driven Reasoning for Horn-$\mathcal{SHIQ}$ Ontologies”. In: **P. of the 21st Int. Joint Conf. on AI (IJCAI 2009)**.
- Krötzsch, Markus, Maximilian Marx, and Sebastian Rudolph (2019). “The Power of the Terminating Chase (Invited Talk)”. In: **P. of the 22nd Int. Conf. on Database Theory (ICDT) 2019)**.
- Krötzsch, Markus, Sebastian Rudolph, and Pascal Hitzler (2013). “Complexities of Horn Description Logics”. In: **ACM Trans. Comput. Log.**