PRACTICAL USES OF EXISTENTIAL RULES IN KNOWLEDGE REPRESENTATION

Part 3: Implementing a Calculus for Horn-$\mathcal{ALC}$ using 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

ECAI, September 4, 2020

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

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

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

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:

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

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.

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

Definition. We define the semantics of Horn-$\mathcal{ALC}$ axioms via translation into equivalent first-order logic formulas:

$$
\begin{aligned}
A \sqsubseteq \bot \quad &\mapsto \quad \forall x.(A(x) \to \bot) \\
\top \sqsubseteq B \quad &\mapsto \quad \forall x.B(x) \\
A \sqsubseteq B \quad &\mapsto \quad \forall x.(A(x) \to B(x)) \\
A \sqcap E \sqsubseteq B \quad &\mapsto \quad \forall x.(A(x) \land E(x) \to B(x)) \\
\exists R.A \sqsubseteq B \quad &\mapsto \quad \forall x.(R(x, y) \land A(y) \to B(x)) \\
A \sqsubseteq \forall R.B \quad &\mapsto \quad \forall x.(A(x) \land R(x, y) \to B(y)) \\
A \sqsubseteq \exists R.B \quad &\mapsto \quad \forall x.(A(x) \to \exists y.R(x, y) \land B(y))
\end{aligned}
$$

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

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

Definition. We define the semantics of Horn-$\mathcal{ALC}$ axioms via translation into equivalent first-order logic formulas:

$$
\begin{aligned}
A \sqsubseteq \bot \quad &\mapsto \quad \forall x.(A(x) \to \bot) \\
\top \sqsubseteq B \quad &\mapsto \quad \forall x.B(x) \\
A \sqsubseteq B \quad &\mapsto \quad \forall x.(A(x) \to B(x)) \\
A \sqcap E \sqsubseteq B \quad &\mapsto \quad \forall x.(A(x) \land E(x) \to B(x)) \\
\exists R.A \sqsubseteq B \quad &\mapsto \quad \forall x.(R(x, y) \land A(y) \to B(x)) \\
A \sqsubseteq \forall R.B \quad &\mapsto \quad \forall x.(A(x) \land R(x, y) \to B(y)) \\
A \sqsubseteq \exists R.B \quad &\mapsto \quad \forall x.(A(x) \to \exists y.R(x, y) \land B(y))
\end{aligned}
$$

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

Often, we remove universal quantifiers from first-order logic formulas.

A Consequence-Based Calculus to Solve Classification

$R^C_A)\ \dfrac{}{A \sqsubseteq A} : A \in \mathsf{Concepts}(\mathcal{O})$	$R^+_\exists)\ \dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.B} : A \sqsubseteq \exists R.B \in \mathcal{O}$
$R^\exists_A)\ \dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D}}{\mathbb{D} \sqsubseteq D} : D \in \mathbb{D}$	$R^-_\exists)\ \dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \exists R.A \sqsubseteq B \in \mathcal{O}$
$R^1_\sqcap)\ \dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \top \sqsubseteq B \in \mathcal{O}$	$R^\bot_\exists)\ \dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq \bot}{\mathbb{C} \sqsubseteq \bot}$
$R^1_\sqcap)\ \dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : A \sqsubseteq B \in \mathcal{O}$	$R_\forall)\ \dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.(\mathbb{D} \sqcap B)} : A \sqsubseteq \forall R.B \in \mathcal{O}$
$R^2_\sqcap)\ \dfrac{\mathbb{C} \sqsubseteq A \quad \mathbb{C} \sqsubseteq E}{\mathbb{C} \sqsubseteq B} : A \sqcap E \sqsubseteq B \in \mathcal{O}$	

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. Original calculus by (Kazakov 2009).

Consequence-Based Calculus: Soundness

Soundness. Show via induction that each rule only produces sound inferences.

For instance, let us show that the following production rule is indeed sound:

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

Proof:

1. By IH: $\mathcal{O} \models \bigwedge_{C \in \mathbb{C}} C(x) \to \exists y.(R(x,y) \land \bigwedge_{D \in \mathbb{D}} D(y))$
2. By IH: $\mathcal{O} \models \bigwedge_{C \in \mathbb{C}} C(x) \to A(x)$
3. By the precondition of the rule: $\mathcal{O} \models A(x) \land R(x,y) \to B(y)$
4. By (1–3) and the semantics of first-order logic:
   $\mathcal{O} \models \bigwedge_{C \in \mathbb{C}} C(x) \to \exists y.(R(x,y) \land \bigwedge_{D \in \mathbb{D}} D(y) \land B(y))$

Consequence-Based Calculus: Completeness

To show **completeness**, we verify the following theorem:

Theorem. If an axiom of the form $A \sqsubseteq B$ is not derived by the previously proposed calculus on input $\mathcal{O}$, then $\mathcal{O} \not\models A \sqsubseteq B$.

Proof Sketch: Using the output of the calculus on input $\mathcal{O}$, we can construct a model for this ontology that contains an element that is in the domain of A but not in the domain of B. Therefore, $\mathcal{O} \not\models A \sqsubseteq B$.

Remark. For a complete proof, check the following references:

- (Kazakov 2009)
- (Simancik, Kazakov, and Horrocks 2011)

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, the calculus may produce at most

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

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

Implementing the Consequence-Based 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.

Proof:

1. Consider a Datalog rule set $\mathcal{R}$, a fact set $\mathcal{F}$, and a fact φ.
2. Let $\mathcal{R}'$ be the grounding of $\mathcal{R}$ using the constants in $\mathcal{F}$.
3. We have that $\mathcal{R}' \cup \mathcal{F} \models \varphi$ if and only if $\mathcal{R} \cup \mathcal{F} \models \varphi$.
4. Checking if $\mathcal{R}' \cup \mathcal{F} \models \varphi$ can be reduced to fact entailment over propositional logic, which can be solved in polynomial time.
5. If $\mathcal{R}$ is fixed, then $\mathcal{R}'$ is polynomial in the number of constants in $\mathcal{F}$.
6. By (3) and (4): if $\mathcal{R}$ is fixed, we can decide if $\mathcal{R} \cup \mathcal{F} \models \varphi$ in polynomial time.

Implementing the Consequence-Based 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 done with the $\mathcal{EL}$ classification calculus). Then:

1. By Theorem 3.4: 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 (↯)

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!

A Horn-$\mathcal{ALC}$ Classification Calculus with Datalog(S)

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

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

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

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.

A Horn-$\mathcal{ALC}$ Classification Calculus with 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:

$$A \sqsubseteq \bot \in \mathcal{O} \mapsto \mathsf{ax}_{\sqsubseteq}(c_A, c_\bot) \qquad \exists R.A \sqsubseteq B \in \mathcal{O} \mapsto \mathsf{ax}_{\exists\sqsubseteq}(c_A, c_R, c_B)$$

$$\top \sqsubseteq B \in \mathcal{O} \mapsto \mathsf{ax}_{\sqsubseteq}(c_\top, c_B) \qquad A \sqsubseteq \forall R.B \in \mathcal{O} \mapsto \mathsf{ax}_{\sqsubseteq\forall}(c_A, c_R, c_B)$$

$$A \sqsubseteq B \in \mathcal{O} \mapsto \mathsf{ax}_{\sqsubseteq}(c_A, c_B) \qquad A \sqsubseteq \exists R.B \in \mathcal{O} \mapsto \mathsf{ax}_{\sqsubseteq\exists}(c_A, c_R, c_B)$$

$$A \sqcap E \sqsubseteq B \in \mathcal{O} \mapsto \mathsf{ax}_{\sqcap\sqsubseteq}(c_A, c_E, c_B) \qquad A \in \mathsf{Concepts}(\mathcal{O}) \mapsto \mathsf{Concept}(c_E)$$

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.

A Horn-$\mathcal{ALC}$ Classification Calculus with Datalog(S)

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

R^C_A)	$\dfrac{}{A \sqsubseteq A} : A \in \text{Concepts}(\mathcal{O})$	$\text{Concept}(x) \rightarrow \text{SC}(\{x\}, x)$
$R^\exists_A$)	$\dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D}}{\mathbb{D} \sqsubseteq D} : D \in \mathbb{D}$	$\text{Ex}(C, r, D) \wedge d \in D \rightarrow \text{SC}(D, d)$
$R^0_\sqcap$)	$\dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \top \sqsubseteq B \in \mathcal{O}$	$\text{SC}(C, a) \wedge \text{ax}_\sqsubseteq(c_\top, b) \rightarrow \text{SC}(C, b)$
$R^1_\sqcap$)	$\dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : A \sqsubseteq B \in \mathcal{O}$	$\text{SC}(C, a) \wedge \text{ax}_\sqsubseteq(a, b) \rightarrow \text{SC}(C, b)$
$R^2_\sqcap$)	$\dfrac{\mathbb{C} \sqsubseteq A \quad \mathbb{C} \sqsubseteq E}{\mathbb{C} \sqsubseteq B} : A \sqcap E \sqsubseteq B \in \mathcal{O}$	$\text{SC}(C, a) \wedge \text{SC}(C, e) \wedge \text{ax}_{\sqcap\sqsubseteq}(a, e, b) \rightarrow \text{SC}(C, b)$

A Horn-$\mathcal{ALC}$ Classification Calculus with Datalog(S)

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

$\mathsf{R}^{+}_{\exists}$)	$\dfrac{\mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.B} : A \sqsubseteq \exists R.B \in \mathcal{O}$	$\mathsf{SC}(C, a) \wedge \mathsf{ax}_{\sqsubseteq\exists}(a, r, b) \rightarrow \mathsf{Ex}(C, r, \{B\})$
$\mathsf{R}^{-}_{\exists}$)	$\dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq A}{\mathbb{C} \sqsubseteq B} : \exists R.A \sqsubseteq B \in \mathcal{O}$	$\mathsf{Ex}(C, r, D) \wedge \mathsf{SC}(D, a) \wedge \mathsf{ax}_{\exists\sqsubseteq}(r, a, b) \rightarrow \mathsf{SC}(C, b)$
$\mathsf{R}^{\bot}_{\exists}$)	$\dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{D} \sqsubseteq \bot}{\mathbb{C} \sqsubseteq \bot}$	$\mathsf{Ex}(C, r, D) \wedge \mathsf{SC}(D, c_{\bot}) \rightarrow \mathsf{SC}(C, c_{\bot})$
$\mathsf{R}^{\bot}_{\forall}$)	$\dfrac{\mathbb{C} \sqsubseteq \exists R.\mathbb{D} \quad \mathbb{C} \sqsubseteq A}{\mathbb{C} \sqsubseteq \exists R.(\mathbb{D} \sqcap B)} : A \sqsubseteq \forall R.B$	$\mathsf{Ex}(C, r, D) \wedge \mathsf{SC}(C, a) \wedge \mathsf{ax}_{\sqsubseteq\forall}(a, r, b) \rightarrow \mathsf{Ex}(C, r, D \cup \{b\})$

A Horn-$\mathcal{ALC}$ Classification Calculus with Datalog(S)

Definition. Let $\mathcal{R}_{\mathsf{HALC}}$ be the following Datalog(S) rule set:

$$\mathsf{Concept}(x) \to \mathsf{SC}(\{x\}, x) \qquad \mathsf{SC}(C, a) \wedge \mathsf{ax}_{\sqsubseteq}(c_{\top}, b) \to \mathsf{SC}(C, b)$$

$$\mathsf{Ex}(C, r, D) \wedge d \in D \to \mathsf{SC}(D, d) \qquad \mathsf{SC}(C, a) \wedge \mathsf{ax}_{\sqsubseteq}(a, b) \to \mathsf{SC}(C, b)$$

$$\mathsf{SC}(C, a) \wedge \mathsf{SC}(C, e) \wedge \mathsf{ax}_{\sqcap\sqsubseteq}(a, e, b) \to \mathsf{SC}(C, b)$$

$$\mathsf{SC}(C, a) \wedge \mathsf{ax}_{\sqsubseteq\exists}(a, r, b) \to \mathsf{Ex}(C, r, \{B\})$$

$$\mathsf{Ex}(C, r, D) \wedge \mathsf{SC}(D, a) \wedge \mathsf{ax}_{\exists\sqsubseteq}(r, a, b) \to \mathsf{SC}(C, b)$$

$$\mathsf{Ex}(C, r, D) \wedge \mathsf{SC}(D, c_{\bot}) \to \mathsf{SC}(C, c_{\bot})$$

$$\mathsf{Ex}(C, r, D) \wedge \mathsf{SC}(C, a) \wedge \mathsf{ax}_{\sqsubseteq\forall}(a, r, b) \to \mathsf{Ex}(C, r, D \cup \{b\})$$

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 $\mathcal{R}_{\mathsf{HALC}} \cup \mathsf{Facts}(\mathcal{O}) \models \mathsf{SC}(c_A, c_B)$.

Implementing the Classification Calculus with VLog

Alas, VLog does not support Datalog(S) reasoning. There maybe some other rule-based language 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.

Moreover, (Carral et al. 2019) have proposed a translation from Datalog(S) into existential rule programs such that:

- The resulting programs terminate with respect to the restricted chase.
- Fact entailment is preserved.

From Datalog(S) to Existential Rules

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

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

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

From Datalog(S) to Existential Rules

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

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

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

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

From Datalog(S) to Existential Rules

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

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

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

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

$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

From Datalog(S) to Existential Rules

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

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

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

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

eve

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

From Datalog(S) to Existential Rules

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

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

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

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

eve

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

$n_{\emptyset}$

From Datalog(S) to Existential Rules

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

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

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

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

$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

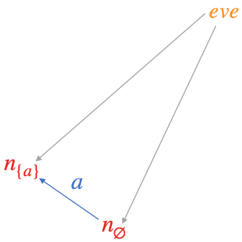
From Datalog(S) to Existential Rules

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

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

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

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



$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

→ $likesAll$

→ SU

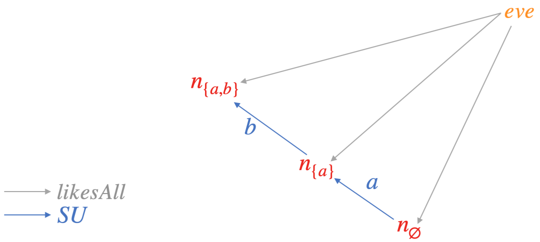
From Datalog(S) to Existential Rules

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

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

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

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



$person(eve)$

$likes(eve, a)$

$likes(eve, b)$

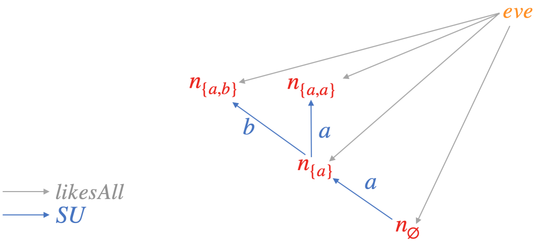
From Datalog(S) to Existential Rules

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

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

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

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



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

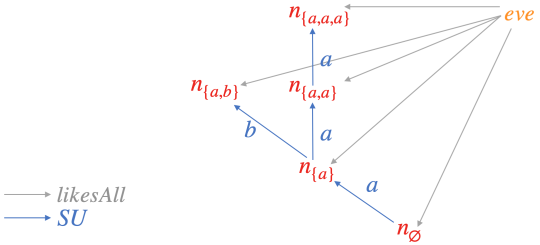
From Datalog(S) to Existential Rules

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

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

$$person(x) \land empty(Y) \to likesAll(x, Y) \quad (1.2)$$

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



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

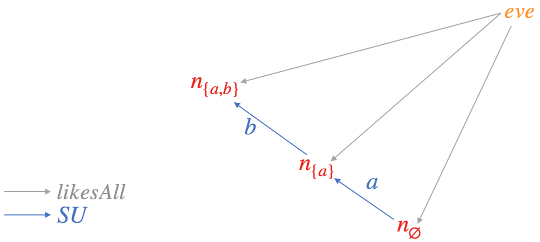
From Datalog(S) to Existential Rules

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

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

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

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



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

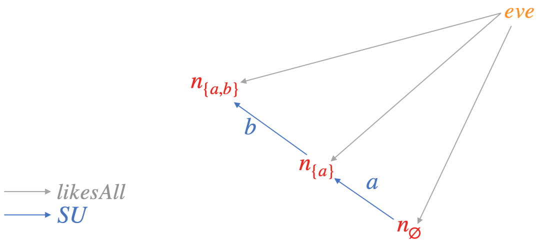
From Datalog(S) to Existential Rules

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

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

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

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



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

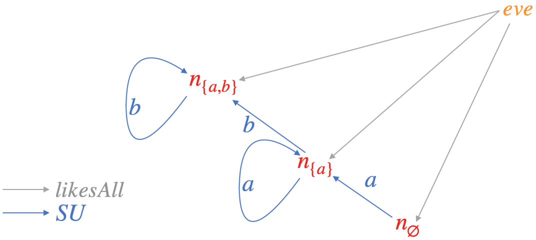
From Datalog(S) to Existential Rules

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

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

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

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



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

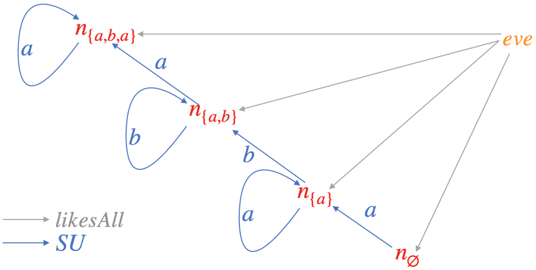
From Datalog(S) to Existential Rules

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

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

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

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



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

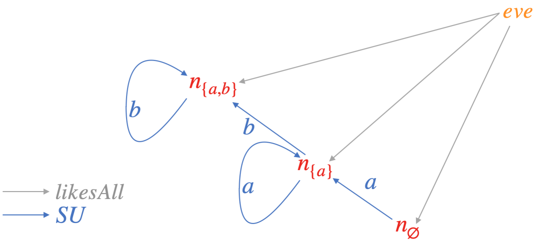
From Datalog(S) to Existential Rules

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

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

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

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



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

From Datalog(S) to Existential Rules

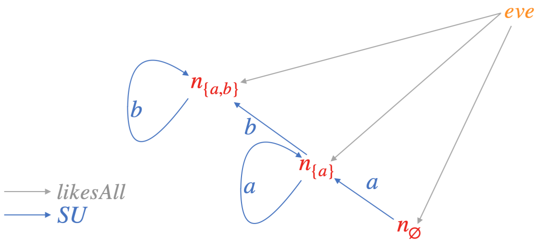
$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, X) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, X \cup \{y\})$

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

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

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

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



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

From Datalog(S) to Existential Rules

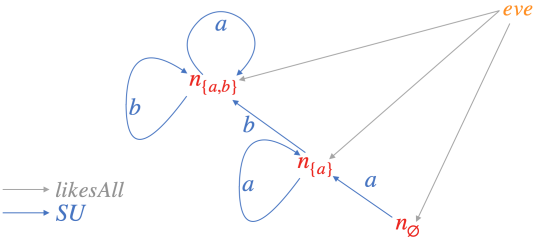
$$\text{Person}(x) \rightarrow \text{LikesAll}(x, \emptyset) \qquad \text{LikesAll}(x, X) \wedge \text{Likes}(x, y) \rightarrow \text{LikesAll}(x, X \cup \{y\})$$

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

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

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

$$SU(U, x, V) \wedge SU(U, y, U) \rightarrow SU(V, y, V) \tag{2.2}$$



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

Experimental Evaluation: Solving Classification

ID	#Ax.	#Set	#SC	#Ex	VLog	Konclude
00040	223K	2K	1051K	334K	432s	5s
00048	142K	19	718K	171K	387s	3s
00477	318K	0	162K	167K	1s	3s
00533	159K	0	965K	351K	132s	2s
00786	152K	12K	2283K	978K	549s	14s

Figure: Ontologies and results for classification (A) showing: axiom count; number of non-singleton “set terms” introduced (#Set); number of SC and Ex facts derived; reasoning time in VLog and Konclude

Experimental Evaluation: Assertion Retrieval

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

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

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

In the above; A, B, and E are concept names (i.e., unary predicates); and R is a role name (i.e., binary predicate).

Definition. Assertion Retrieval is the reasoning task of computing all axioms of the form $A(a)$ or $R(a,b)$ that ate 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 assertion retrieval.

Experimental Evaluation: Assertion Retrieval

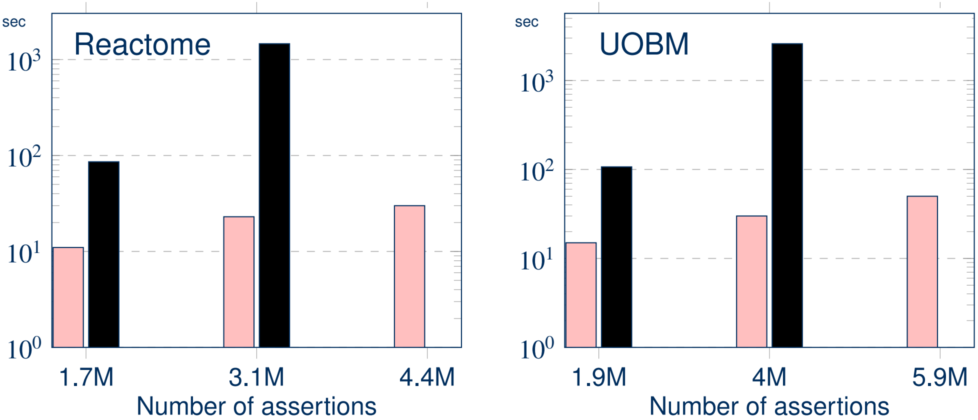


Figure: Experimental results for class retrieval (B) in VLog (pink/grey) and Konclude (black); note the log scale

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

Hands on Session!

References I

- 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.**
- Simancik, Frantisek, Yevgeny Kazakov, and Ian Horrocks (2011). “Consequence-Based Reasoning beyond Horn Ontologies”. In: **P. of the 22nd Int. Joint Conf. on AI (IJCAI 2011)**.