MORe: A Modular OWL Reasoner for Ontology Classification

Ana Armas Romero, Bernardo Cuenca Grau,
Ian Horrocks, Ernesto Jiménez Ruiz

Department of Computer Science, University of Oxford

July 2013

OWL 2 REASONERS
VS
PROFILE SPECIFIC REASONERS

- OWL 2 reasoners
  HermiT, Pellet, Fact++, JFact, RacerPro...
  - Complete for OWL 2
  - Highly optmized, but still too slow for some ontologies.

OWL 2 Reasoners vs Profile Specific Reasoners

- OWL 2 reasoners
  HermiT, Pellet, Fact++, JFact, RacerPro...
  - Complete for OWL 2
  - Highly optmized, but still too slow for some ontologies.

- Profile specific reasoners
  ELK, CEL (EL), Jena (RL), OWLIM (RL, QL)
  - Extremely efficient and scalable
  - No completeness guarantee if ontology contains even just a few axioms outside relevant fragment.

OWL 2 Reasoner
+
Profile Specific Reasoner

MORe integrates an OWL 2 Reasoner and an EL reasoner
$\longrightarrow$ currently HermiT/JFact and ELK
finding two subsets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{O}$ such that

OWL 2 Reasoner
+
Profile Specific Reasoner

MORe integrates an OWL 2 Reasoner and an EL reasoner

$\longrightarrow$ currently HermiT/JFact and ELK

finding two subsets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{O}$ such that

- ELK classifies Sig$(\mathcal{M}_1)$ with axioms in $\mathcal{M}_1$

OWL 2 REASONER
+
PROFILE SPECIFIC REASONER

MORe integrates an OWL 2 Reasoner and an EL reasoner
$\longrightarrow$ currently HermiT/JFact and ELK
finding two subsets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{O}$ such that

- ELK classifies Sig$(\mathcal{M}_1)$ with axioms in $\mathcal{M}_1$
- OWL 2 reasoner classifies Sig$(\mathcal{M}_2)$ with axioms in $\mathcal{M}_2$

OWL 2 Reasoner + Profile Specific Reasoner

MORe integrates an OWL 2 Reasoner and an EL reasoner
$\longrightarrow$ currently HermiT/JFact and ELK
finding two subsets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{O}$ such that

- ELK classifies Sig$(\mathcal{M}_1)$ with axioms in $\mathcal{M}_1$
- OWL 2 reasoner classifies Sig$(\mathcal{M}_2)$ with axioms in $\mathcal{M}_2$
- $\mathcal{M}_2$ is as small as possible
  —reduce workload of OWL 2 reasoner!

OWL 2 Reasoner + Profile Specific Reasoner

MORe integrates an OWL 2 Reasoner and an EL reasoner
$\longrightarrow$ currently HermiT/JFact and ELK
finding two subsets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{O}$ such that

- ELK classifies Sig$(\mathcal{M}_1)$ with axioms in $\mathcal{M}_1$
- OWL 2 reasoner classifies Sig$(\mathcal{M}_2)$ with axioms in $\mathcal{M}_2$
- $\mathcal{M}_2$ is as small as possible
  —reduce workload of OWL 2 reasoner!
- Sig$(\mathcal{M}_1) \cup$ Sig$(\mathcal{M}_2) =$ Sig$(\mathcal{O})$
  —but never lose completenes!!

Modules as glue for reasoners

- A module is a subset of an ontology that preserves entailments over a given signature Σ
- Modules based on syntactic locality can be extracted in polynomial time
- $\bot$-modules (based on $\bot$-locality) have a special property:

Modules as glue for reasoners

- A module is a subset of an ontology that preserves entailments over a given signature Σ
- Modules based on syntactic locality can be extracted in polynomial time
- $\bot$-modules (based on $\bot$-locality) have a special property:

If $A \in \Sigma$ and $\mathcal{O} \models A \sqsubseteq B$
then
$\mathcal{M}^{\bot}_{\mathcal{O},\Sigma} \models A \sqsubseteq B$

Modules as glue for reasoners

- A module is a subset of an ontology that preserves entailments over a given signature Σ
- Modules based on syntactic locality can be extracted in polynomial time
- $\bot$-modules (based on $\bot$-locality) have a special property:

$$\text{If } A \in \Sigma \text{ and } \mathcal{O} \models A \sqsubseteq B$$

then

$$\mathcal{M}^{\bot}_{\mathcal{O},\Sigma} \models A \sqsubseteq B$$

even if B wasn't in Σ!

$\bot$-MODULES IN ACTION!

MORe integrates an OWL 2 Reasoner and an EL reasoner
$\longrightarrow$ currently HermiT/JFact and ELK

finding two subsets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{O}$ such that

- ELK classifies Sig$(\mathcal{M}_1)$ with $\mathcal{M}_1$
- OWL 2 reasoner classifies Sig$(\mathcal{M}_2)$ with $\mathcal{M}_2$
- $\mathcal{M}_2$ is as small as possible
  —reduce workload of OWL 2 reasoner!
- Sig$(\mathcal{M}_1) \cup$ Sig$(\mathcal{M}_2) =$ Sig$(\mathcal{O})$
  —but never lose completenes!!

$\bot$-MODULES IN ACTION!

MORe integrates an OWL 2 Reasoner and an EL reasoner

$\longrightarrow$ currently HermiT/JFact and ELK

finding a subset $\Sigma_{\text{EL}} \subseteq \text{Sig}(\mathcal{O})$ such that

- ELK classifies Σ_{EL} with $\mathcal{M}^{\bot}_{\mathcal{O},\Sigma_{\text{EL}}}$
- OWL 2 reasoner classifies $\overline{\Sigma_{\text{EL}}} = \text{Sig}(\mathcal{O}) \setminus \Sigma_{\text{EL}}$ with $\mathcal{M}^{\bot}_{\mathcal{O},\overline{\Sigma_{\text{EL}}}}$
- $\mathcal{M}^{\bot}_{\mathcal{O},\overline{\Sigma_{\text{EL}}}}$ is as small as possible
  —reduce workload of OWL 2 reasoner!
- $\Sigma_{\text{EL}} \cup \overline{\Sigma_{\text{EL}}}$
  —but never lose completenes!!

EL-SIGNATURES

We call $\Sigma_{\mathsf{EL}} \subseteq \mathsf{Sig}(\mathcal{O})$ an EL-signature for $\mathcal{O}$ if $\mathcal{M}^{\perp}_{[\mathcal{O},\Sigma_{\mathsf{EL}}]}$ is in EL.

EL-SIGNATURES

We call $\Sigma_{\mathsf{EL}} \subseteq \mathrm{Sig}(\mathcal{O})$ an EL-signature for $\mathcal{O}$ if $\mathcal{M}^{\perp}_{[\mathcal{O},\Sigma_{\mathsf{EL}}]}$ is in EL.

Computing an EL-signature is like extractracting a module - but backwards!

- Computing a module: start with a signature Σ, obtain the subset of "axioms for that signature" in $\mathcal{O}$.
- Computing an EL-signature: start with a set $\mathcal{O}' \subseteq \mathcal{O}$ of axioms that we DON'T want, obtain a signature (whose $\perp$-module contains no axioms from $\mathcal{O}'$)

MORe does not always compute maximal EL signatures, but it computes fairly large ones very fast.

DISCUSSION

- EL-signatures obtained typically large when most axioms are in EL.
  $\longrightarrow$ developed heuristics that seem to lead to larger EL-signatures in most cases

- Integrated reasoners are used as black boxes:
  any OWL reasoner, and
  any EL reasoner
  could be integrated in MORe's infrastructure as is —and only minor alterations would be needed to integrate a reasoner for a different profile.

Experimental Results

	Expressivity	$\|\mathrm{Sig}(\mathcal{O})\|$	$\|\mathcal{O}\|$	$\|\mathcal{O} \setminus \mathcal{O}_{\mathsf{ELK}}\|$	$\|\mathcal{M}_{\mathsf{OWL2}}\|$
Gazetteer	$\mathcal{ALE}+$	517,039	652,361	0	0%
Cardiac Electrophys.	$\mathcal{SHF}(\mathcal{D})$	81,020	124,248	22	1%
Protein	$\mathcal{S}$	35,205	46,114	15	22%
Biomodels	$\mathcal{SRIF}$	187,577	439,248	22,104	45%
Cell Cycle v0.95	$\mathcal{SRI}$	144,619	511,354	1	$<$0.1%
Cell Cycle v2.01	$\mathcal{SRI}$	106,517	624,702	9	98%
NCI v09.12d	$\mathcal{SH}(\mathcal{D})$	77,571	109,013	4,682	58%
NCI v13.03d	$\mathcal{SH}(\mathcal{D})$	97,652	136,902	158	57%
$\mathrm{SNOMED}_{15\sqcup}$	$\mathcal{ALCR}$	291,216	291,185	15	3%
SNOMED+LUCADA	$\mathcal{ALCRIQ}(\mathcal{D})$	309,405	550,453	122	0.1%

Experimental Results

	$\text{MORe}_{\text{HermiT}}$		HermiT	$\text{MORe}_{\text{Pellet}}$		Pellet
	HermiT	total		Pellet	total	
Gazetteer	0	20.6	651	0	20.3	1,414
Cardiac Electrophys.	0.3	6.3	22.7	0.3	5.5	11.0
Protein	2.0	4.8	10.0	2.0	4.7	2,920
Biomodels	377	487	582	373	483	1,915
Cell Cycle v0.95	$<$0.1	9.9	mem	$<$0.1	10.4	3,433
Cell Cycle v2.01	mem	mem	mem	mem	mem	3,435
NCI v09.12d	244	252	261	256	266	93.6
NCI v13.03d	45.1	62.7	68.4	45.7	62.9	191
$\text{SNOMED}_{15\sqcup}$	4.5	25.4	1,395	4.4	22.9	4,314
SNOMED+LUCADA	1.1	28.8	1,302	1.2	29.2	mem

Ongoing Work

Currently developing a new algorithm that should reduce the workload of the OWL reasoner even further by computing

$\longrightarrow$ a lower and upper bound for the classification

and

$\longrightarrow$ a very reduced set of axioms enough to check the dubious subsumption relations

$\sim$ alternative notion of module, wouldn't preserve all kinds of entailments, only subsumption between atomic classes

Thanks!