Which Kind of Module Should I Extract?

Uli Sattler[1] *Thomas Schneider*[1] Michael Zakharyaschev[2]

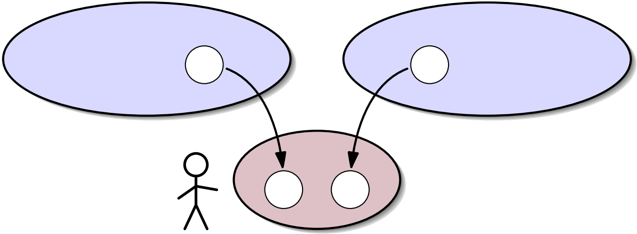
[1]School of Computer Science, University of Manchester
[2]Birkbeck College, London

DL, 28 July 2009

And now . . .

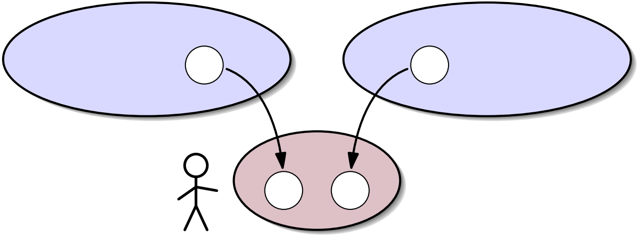
Why module extraction?

Reuse external ontologies: borrow knowledge about certain terms



Why module extraction?

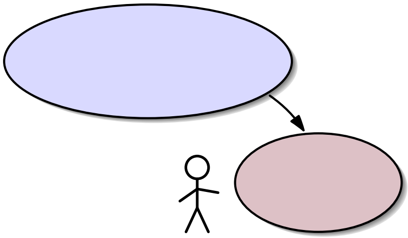
Reuse external ontologies: borrow knowledge about certain terms



- Provides access to well-established knowledge
- Doesn't require expertise in external disciplines

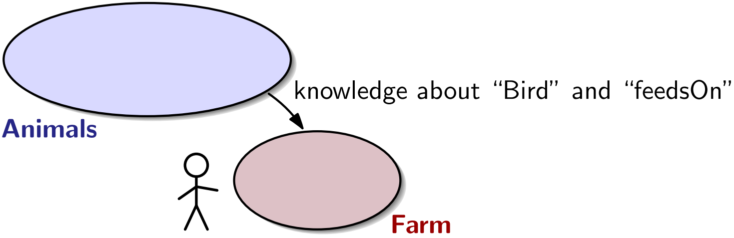
Why module extraction?

Reuse external ontologies: borrow knowledge about certain terms



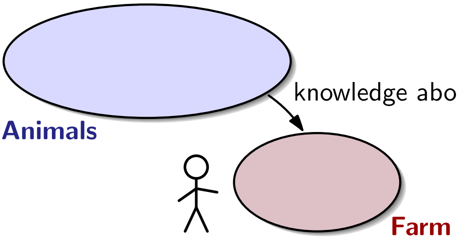
Why module extraction?

Reuse external ontologies: borrow knowledge about certain terms



Why module extraction?

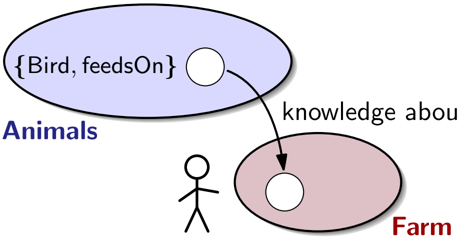
Reuse external ontologies: borrow knowledge about certain terms



How much of **Animals** do we need?

Why module extraction?

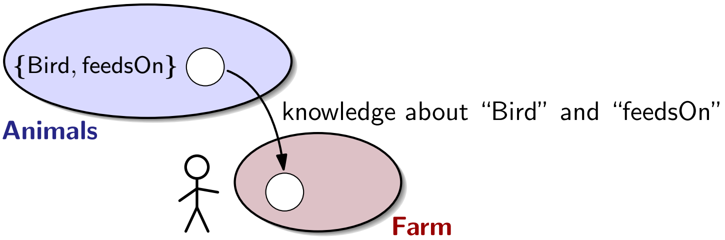
Reuse external ontologies: borrow knowledge about certain terms



How much of **Animals** do we need?

Why module extraction?

Reuse external ontologies: borrow knowledge about certain terms

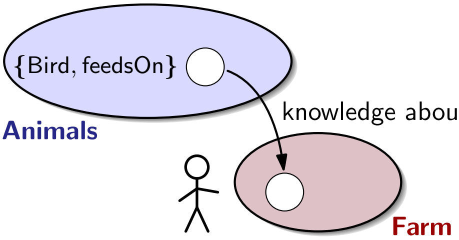


Coverage Import *everything* relevant for the chosen terms.

Economy Import *only* what's relevant for them.
Compute that module quickly.

Why module extraction?

Reuse external ontologies: borrow knowledge about certain terms



Coverage Import *everything* relevant for the chosen terms.

Economy Import *only* what's relevant for them.
Compute that module quickly.

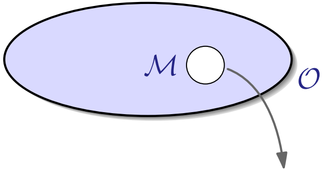
Modules that provide coverage

Input Ontology $\mathcal{O}$ — a set of axioms
Signature Σ (set of concept and role names from $\mathcal{O}$)

Output **a Σ-module $\mathcal{M}$ of $\mathcal{O}$:**

- $\mathcal{M} \subseteq \mathcal{O}$
- $\mathcal{M}$ and $\mathcal{O}$ have the same Σ-entailments:

  For all axioms α using only terms from Σ,
  $\mathcal{O} \models \alpha$ iff $\mathcal{M} \models \alpha$



Modules that provide coverage

Input Ontology $\mathcal{O}$ — a set of axioms
Signature Σ (set of concept and role names from $\mathcal{O}$)

Output **a Σ-module $\mathcal{M}$ of $\mathcal{O}$**:

- $\mathcal{M} \subseteq \mathcal{O}$
- $\mathcal{M}$ and $\mathcal{O}$ have the same Σ-entailments:

  For all axioms α using only terms from Σ,
  $\mathcal{O} \models \alpha$ iff $\mathcal{M} \models \alpha$

Coverage ✓

Modules that provide coverage

Input Ontology $\mathcal{O}$ — a set of axioms
Signature Σ (set of concept and role names from $\mathcal{O}$)

Output **a Σ-module $\mathcal{M}$ of $\mathcal{O}$**:

- $\mathcal{M} \subseteq \mathcal{O}$
- $\mathcal{M}$ and $\mathcal{O}$ have the same Σ-entailments:

  For all axioms α using only terms from Σ,
  $\mathcal{O} \models \alpha$ iff $\mathcal{M} \models \alpha$

Coverage ✓

Economy Minimality $\overset{!}{\longleftrightarrow}$ efficient computability

Modules that provide coverage

Input Ontology $\mathcal{O}$ — a set of axioms
Signature Σ (set of concept and role names from $\mathcal{O}$)

Output **a Σ-module $\mathcal{M}$ of $\mathcal{O}$**:

- $\mathcal{M} \subseteq \mathcal{O}$
- $\mathcal{M}$ and $\mathcal{O}$ have the same Σ-entailments:

  For all axioms α using only terms from Σ,
  $\mathcal{O} \models \alpha$ iff $\mathcal{M} \models \alpha$

Coverage ✓

Economy Minimality $\overset{!}{\longleftrightarrow}$ efficient computability

conservativity-based modules — locality-based modules

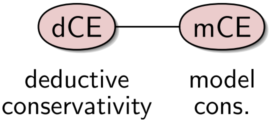
Relevant module types

dCE

deductive conservativity

intractable . . . undecidable

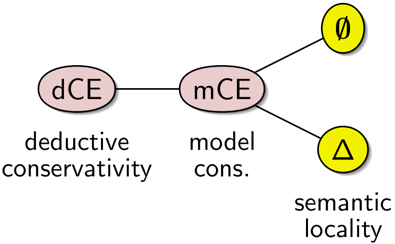
Relevant module types



x — y x-module$(\mathcal{O}, \Sigma) \subseteq y$-module$(\mathcal{O}, \Sigma)$

intractable . . . undecidable

Relevant module types

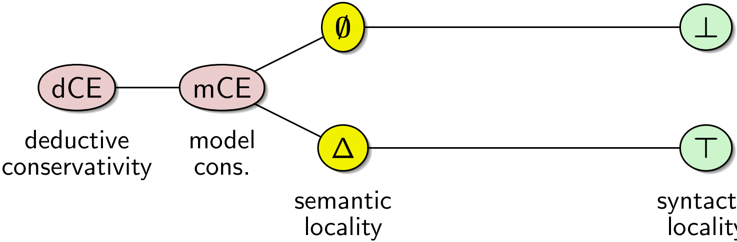


x — y x-module$(\mathcal{O}, \Sigma) \subseteq y$-module$(\mathcal{O}, \Sigma)$

(pink) intractable . . . undecidable

(yellow) as difficult as reasoning

Relevant module types



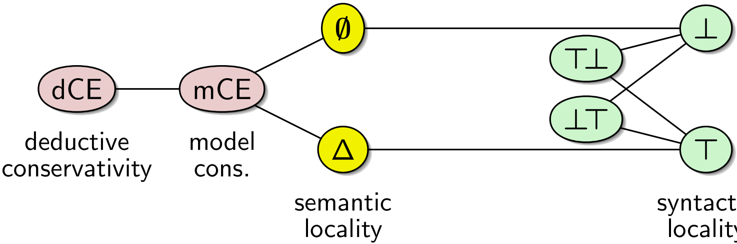
x — y x-module$(\mathcal{O}, \Sigma) \subseteq y$-module$(\mathcal{O}, \Sigma)$

intractable . . . undecidable

as difficult as reasoning

tractable

Relevant module types



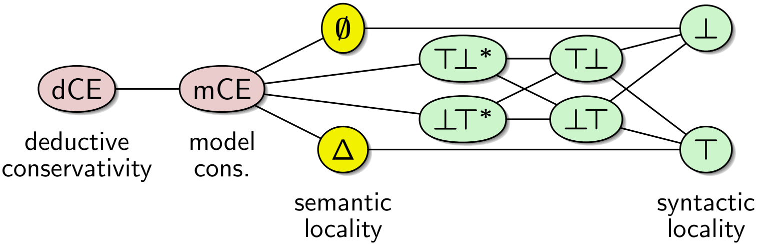
x — y x-module$(\mathcal{O}, \Sigma) \subseteq y$-module$(\mathcal{O}, \Sigma)$

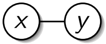
intractable . . . undecidable

as difficult as reasoning

tractable

Relevant module types





x-module$(\mathcal{O}, \Sigma) \subseteq y$-module$(\mathcal{O}, \Sigma)$

intractable . . . undecidable

as difficult as reasoning

tractable

Goals

- General framework for comparing module notions that provide coverage
- Identify relevant properties
- Application to conservativity-based and locality-based modules

And now . . .

Intuitions

- $\mathcal{O}_1$ and $\mathcal{O}_2$ are inseparable w.r.t. Σ:
  The knowledge about Σ in $\mathcal{O}_1$ and $\mathcal{O}_2$ can't be distinguished
- Different degrees of distinguishability

Intuitions

- $\mathcal{O}_1$ and $\mathcal{O}_2$ are inseparable w.r.t. Σ:
  The knowledge about Σ in $\mathcal{O}_1$ and $\mathcal{O}_2$ can't be distinguished
- Different degrees of distinguishability
- Notation: $\mathcal{O}_1 \equiv^S_\Sigma \mathcal{O}_2$
- $\equiv^S_\Sigma$ is an equivalence relation

Intuitions

- $\mathcal{O}_1$ and $\mathcal{O}_2$ are inseparable w.r.t. Σ:
  The knowledge about Σ in $\mathcal{O}_1$ and $\mathcal{O}_2$ can't be distinguished
- Different degrees of distinguishability
- Notation: $\mathcal{O}_1 \equiv_{\Sigma}^{S} \mathcal{O}_2$
- $\equiv_{\Sigma}^{S}$ is an equivalence relation
- Inseparability relation: $S = \{\equiv_{\Sigma}^{S} \mid \Sigma \text{ is a signature}\}$

Different inseparability relations

- $\mathcal{O}_1 \equiv_{\Sigma}^{\mathsf{dCE}} \mathcal{O}_2$ if:
  $\mathcal{O}_1$ and $\mathcal{O}_2$ entail the same Σ-concept subsumptions

Different inseparability relations

- $\mathcal{O}_1 \equiv_{\Sigma}^{\mathsf{dCE}} \mathcal{O}_2$ if:

  $\mathcal{O}_1$ and $\mathcal{O}_2$ entail the same Σ-concept subsumptions

- $\mathcal{O}_1 \equiv_{\Sigma}^{\mathsf{mCE}} \mathcal{O}_2$ if:

  $\mathcal{O}_1$ and $\mathcal{O}_2$ have the same models w.r.t. Σ

Different inseparability relations

- $\mathcal{O}_1 \equiv_{\Sigma}^{\mathsf{dCE}} \mathcal{O}_2$ if:

  $\mathcal{O}_1$ and $\mathcal{O}_2$ entail the same Σ-concept subsumptions

- $\mathcal{O}_1 \equiv_{\Sigma}^{\mathsf{mCE}} \mathcal{O}_2$ if:

  $\mathcal{O}_1$ and $\mathcal{O}_2$ have the same models w.r.t. Σ

- $\mathcal{O}_1 \equiv_{\Sigma}^{\perp} \mathcal{O}_2$ if:

  $\mathcal{O}_1$ and $\mathcal{O}_2$ have the same $\perp$-module w.r.t. Σ

Different inseparability relations

- $\mathcal{O}_1 \equiv_\Sigma^{\mathrm{dCE}} \mathcal{O}_2$ if:
  $\mathcal{O}_1$ and $\mathcal{O}_2$ entail the same Σ-concept subsumptions
- $\mathcal{O}_1 \equiv_\Sigma^{\mathrm{mCE}} \mathcal{O}_2$ if:
  $\mathcal{O}_1$ and $\mathcal{O}_2$ have the same models w.r.t. Σ
- $\mathcal{O}_1 \equiv_\Sigma^{\bot} \mathcal{O}_2$ if:
  $\mathcal{O}_1$ and $\mathcal{O}_2$ have the same $\bot$-module w.r.t. Σ

Analogous definition for

$\equiv_\Sigma^{\emptyset}$ $\equiv_\Sigma^{\Delta}$ $\equiv_\Sigma^{\top}$ $\equiv_\Sigma^{\top\bot}$ $\equiv_\Sigma^{\bot\top}$ $\equiv_\Sigma^{\top\bot^*}$ $\equiv_\Sigma^{\bot\top^*}$

Inseparability relations induce modules

Let S be an inseparability relation, Σ a signature and $\mathcal{M} \subseteq \mathcal{O}$.

$\mathcal{M}$ is called	if	see
an S_Σ-*module of* $\mathcal{O}$	$\mathcal{M} \equiv^S_\Sigma \mathcal{O}$	1

Example: $S = \mathsf{dCE}$, $\Sigma = \{\mathsf{Bird}, \mathsf{feedsOn}\}$, $\mathcal{M}$ contains Grass.

1. $\mathcal{O} \models \mathsf{Bird} \sqsubseteq \exists\,\mathsf{feedsOn}.\top$ iff $\mathcal{M} \models \mathsf{Bird} \sqsubseteq \exists\,\mathsf{feedsOn}.\top$

Inseparability relations induce modules

Let S be an inseparability relation, Σ a signature and $\mathcal{M} \subseteq \mathcal{O}$.

$\mathcal{M}$ is called	if	see
an S_Σ*-module of* $\mathcal{O}$	$\mathcal{M} \equiv^S_\Sigma \mathcal{O}$	1
a *self-contained* S_Σ*-module of* $\mathcal{O}$	$\mathcal{M} \equiv^S_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \mathcal{O}$	2

Example: $S = \text{dCE}$, $\Sigma = \{\text{Bird}, \text{feedsOn}\}$, $\mathcal{M}$ contains Grass.

1. $\mathcal{O} \models \text{Bird} \sqsubseteq \exists\, \text{feedsOn}.\top$ iff $\mathcal{M} \models \text{Bird} \sqsubseteq \exists\, \text{feedsOn}.\top$
2. $\mathcal{O} \models \text{Bird} \sqsubseteq \exists\, \text{feedsOn}.\text{Grass}$ iff $\mathcal{M} \models \text{Bird} \sqsubseteq \exists\, \text{feedsOn}.\text{Grass}$

Inseparability relations induce modules

Let S be an inseparability relation, Σ a signature and $\mathcal{M} \subseteq \mathcal{O}$.

$\mathcal{M}$ is called	if	see
an *S_Σ-module of $\mathcal{O}$*	$\mathcal{M} \equiv^S_\Sigma \mathcal{O}$	1
a *self-contained S_Σ-module of $\mathcal{O}$*	$\mathcal{M} \equiv^S_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \mathcal{O}$	2
a *depleting S_Σ-module of $\mathcal{O}$*	$\emptyset \equiv^S_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \mathcal{O} \setminus \mathcal{M}$	3

Example: $S = \mathsf{dCE}$, $\Sigma = \{\mathsf{Bird}, \mathsf{feedsOn}\}$, $\mathcal{M}$ contains Grass.

1. $\mathcal{O} \models \mathsf{Bird} \sqsubseteq \exists\, \mathsf{feedsOn}.\top$ iff $\mathcal{M} \models \mathsf{Bird} \sqsubseteq \exists\, \mathsf{feedsOn}.\top$
2. $\mathcal{O} \models \mathsf{Bird} \sqsubseteq \exists\, \mathsf{feedsOn}.\mathsf{Grass}$ iff $\mathcal{M} \models \mathsf{Bird} \sqsubseteq \exists\, \mathsf{feedsOn}.\mathsf{Grass}$
3. $\mathcal{O} \setminus \mathcal{M}$ entails only tautologies w.r.t. $\{\mathsf{Bird}, \mathsf{feedsOn}, \mathsf{Grass}\}$.

And now . . .

Robustness properties (1)

S is robust under vocabulary restrictions:

If $\mathcal{O}_1 \equiv^S_\Sigma \mathcal{O}_2$ and $\Sigma' \subseteq \Sigma$, then $\mathcal{O}_1 \equiv^S_{\Sigma'} \mathcal{O}_2$.

Robustness properties (1)

S is robust under vocabulary restrictions:

If $\mathcal{O}_1 \equiv^S_\Sigma \mathcal{O}_2$ and $\Sigma' \subseteq \Sigma$, then $\mathcal{O}_1 \equiv^S_{\Sigma'} \mathcal{O}_2$.

Consequences:

If $\mathcal{M}$ is a Σ-module of $\mathcal{O}$ and $\Sigma' \subseteq \Sigma$,
then $\mathcal{M}$ is a Σ'-module of $\mathcal{O}$.

$\leadsto$ On restricting the signature, no new import is necessary.

Robustness properties (2)

- Vocabulary extensions

  If $\mathcal{M}$ is a Σ-module of $\mathcal{O}$ and $(\Sigma' \setminus \Sigma) \cap \mathrm{sig}(\mathcal{O}) = \emptyset$,
  then $\mathcal{M}$ is a Σ'-module of $\mathcal{O}$.

  $\leadsto$ On extending the signature with terms outside $\mathcal{O}$,
  no new import is necessary.

Robustness properties (2)

- Vocabulary extensions

  If $\mathcal{M}$ is a Σ-module of $\mathcal{O}$ and $(\Sigma' \setminus \Sigma) \cap \text{sig}(\mathcal{O}) = \emptyset$, then $\mathcal{M}$ is a Σ'-module of $\mathcal{O}$.

  $\rightsquigarrow$ On extending the signature with terms outside $\mathcal{O}$, no new import is necessary.

- Replacement

  If $\mathcal{M}$ is a Σ-module of $\mathcal{O}$ and $(\text{sig}(\mathcal{O}') \setminus \Sigma) \cap \text{sig}(\mathcal{O}) = \emptyset$, then $\mathcal{M} \cup \mathcal{O}'$ is a Σ-module of $\mathcal{O} \cup \mathcal{O}'$.

  $\rightsquigarrow$ The module relation is compatible with imports.

Robustness properties (2)

- Vocabulary extensions

  If $\mathcal{M}$ is a Σ-module of $\mathcal{O}$ and $(\Sigma' \setminus \Sigma) \cap \mathrm{sig}(\mathcal{O}) = \emptyset$,
  then $\mathcal{M}$ is a Σ'-module of $\mathcal{O}$.

  $\leadsto$ On extending the signature with terms outside $\mathcal{O}$,
  no new import is necessary.

- Replacement

  If $\mathcal{M}$ is a Σ-module of $\mathcal{O}$ and $(\mathrm{sig}(\mathcal{O}') \setminus \Sigma) \cap \mathrm{sig}(\mathcal{O}) = \emptyset$,
  then $\mathcal{M} \cup \mathcal{O}'$ is a Σ-module of $\mathcal{O} \cup \mathcal{O}'$.

  $\leadsto$ The module relation is compatible with imports.

- Joins

  If we have two indistinguishable ontologies,
  it suffices to import one of them.

Overview of properties

Inseparability rel. (IR) / Property	$\equiv_\Sigma^{dCE}$	$\equiv_\Sigma^{mCE}$	$\equiv_\Sigma^{\emptyset}$ $\equiv_\Sigma^{\Delta}$ $\equiv_\Sigma^{\bot}$ $\equiv_\Sigma^{\top}$	$\equiv_\Sigma^{\top\bot}$ $\equiv_\Sigma^{\bot\top}$	$\equiv_\Sigma^{\top\bot*}$ $\equiv_\Sigma^{\bot\top*}$
Modules are induced …					
modules	✓	✓	✓	✗	✓
self-contained modules	✗	✗	✓	✓	✓
depleting modules	✗	✗	✓	✓	✓
IR is robust under …					
vocab. restrictions	✓	✓	✓	✗	✓
vocab. extensions	✗	✓	✓	✗	✓
replacement	✗	✓	✓	✓	✓
joins	✗	✓	✓	✓	✓

And now . . .

Conclusions

- mCE-based and (most) locality-based modules are very robust.
- dCE-based modules are not robust.
- Locality-based modules can be extracted efficiently.

  $\rightsquigarrow$ Intermediate step for extracting mCE-based modules

Conclusions

- mCE-based and (most) locality-based modules are very robust.
- dCE-based modules are not robust.
- Locality-based modules can be extracted efficiently.
  $\rightsquigarrow$ Intermediate step for extracting mCE-based modules

Thank you.