



Языки онтологического моделирования

Курс: Концептуальное
моделирование предметных
областей



План лекции

- Онтология
- Примеры на языках KIF, Ontolingua
- Основные сведения о логиках описаний
- Языки OWL и OWL 2
- Место онтологий в семантическом вебе



Преследуемые цели

- Выработка общего словаря предметной области
 - Взаимодействие с разными исследовательскими группами
 - Взаимодействие специалистов предметных областей и специалистов в информационных технологиях и науке о данных
 - Создание исследовательских сообществ
- Согласование понятий и семантики объектов в предметной области
 - Концептуализация предметной области
 - Формализация определения понятий, ограничений их интерпретации
- Понятные машине спецификации
 - Возможность автоматического вывода
- Пока не говорим о применении



Определение онтологии

- Онтология
 - Явным образом определённая спецификация концептуализации
 - Формальное представление множества понятий предметной области и связей между этими понятиями
 - Словарь предметной области, содержащий точные определения или аксиомы, ограничивающие смысл терминов
 - Онтология = словарь предметной области + логическая теория



Назначение онтологий

- Онтологии используются для
 - Определения состава понятий, используемых в предметной области
 - Описания наиболее общих свойств понятий предметной области
 - Соглашение сообщества об интерпретации понятий
 - Определения общедоступной логической теории предметной области
 - Отнесение объектов к понятиям предметной области
- Онтологии изначально не предназначены для
 - Абстрактного представления данных и их структуры
 - Хранения данных об объекте предметной области
 - Внутренних структур информационных систем



ЯЗЫКИ ОНТОЛОГИЙ

- «Явным образом определённая спецификация»
 - Выраженная в определённой модели данных
- «Понятна машине»
 - Формальная модель
- Разновидности языков онтологий
 - Неформальные
 - Лингвистические (тезаурусы)
 - Связи понятий и указанием силы связи
 - Графические (Семантические сети, RDF, UML)
 - Формальные
 - Логика первого порядка (KIF, Ontolingua)
 - Неклассические логики (F-Logic, модальная логика)
 - Логики описаний (OWL, OWL 2)



Элементы онтологий

- Индивиды, объекты, экземпляры
 - Константы
- Понятия, концепты, классы, типы, термины
 - Унарные предикаты
- Отношения, свойства, роли, связи
 - Бинарные предикаты или предикаты высших порядков
- Утверждения, определения
 - Аксиомы, теоремы (выведенные знания) в модели, определяемой языком онтологий



Язык KIF

- Язык обмена знаниями
- Язык для выражения логики предикатов первого порядка

(forall ?W (\Rightarrow (writer ?W)

(exists (?R ?D) (and

(reader ?R) (document ?D)

(writes ?W ?D) (reads ?R ?D)

(not (understands ?R ?D))))))



Язык Ontolingua

- Классы, отношения, функции, индивиды
(define-class class-name (?instance-variable)
 "documentation string"
 :def or :iff-def KIF-sentence
 :constraints KIF-sentence
 :sufficient KIF-sentence
 :equivalence KIF-sentence
 :axioms KIF-sentence)
- Определение класса Plan
(define-class Plan (?Plan)
 "Plan is the Activity-Spec in the Intended-Purpose Relationship"
 :axioms ((range Sub-Plan-Of Plan) (domain Sub-Plan-Of Plan)
 (instance-of Class Plan) (subclass-of Activity-Spec Plan)
 (subclass-of Qua-Entity Plan))
 :axioms (= (slot-cardinality Plan Intended-Purpose) 1)
 :iff-def (Exists (?Soa) (Intended-Purpose ?Plan ?Soa))
)



Логики описаний (Description logics)

- Семейство языков представления знаний
- Ответвление от семантических сетей и логики KL-ONE
- Описывают предметную область в терминах концептов (классов), ролей (отношений) и индивидов
- Определяют формальную семантику утверждений
- Используют разрешимые фрагменты логики первого порядка для определённых задач
- Предоставляют методы и средства логического вывода
 - Проверка непротиворечивости определений (непустая интерпретация)
 - Проверка поглощения между понятиями
 - Классификация понятий и индивидов на основе их описаний
- Процедура вывода корректна и полна
 - Все выводимые факты корректны в рамках теории
 - Выводятся все возможные корректные факты



Конструкции логик описания

- Элементы дескриптивных логик
 - Атомарные концепты (C)
 - Атомарные роли (\mathcal{R})
 - Индивиды (i)
- Конструкции
 - Пустой концепт \perp
 - Полный концепт \top
 - Пересечение, объединение концептов $C_1 \sqcap C_2, C_1 \sqcup C_2$
 - Дополнение концепта $\neg C$
 - Ограничение существования или всеобщности на значений роли $\exists \mathcal{R}.C, \forall \mathcal{R}.C$
 - Ограничение множественности ролей $\leq_n \mathcal{R}.C, \geq_n \mathcal{R}.C$
 - Пересечение и объединение ролей $\mathcal{R}_1 \sqcap \mathcal{R}_2, \mathcal{R}_1 \sqcup \mathcal{R}_2$
 - Инверсная роль \mathcal{R}^-
 - Принадлежность индивида концепту $i : C$
 - Вложенность концептов $C_1 \sqsubseteq C_2$



Теоретико-множественная интерпретация

- $(\perp)^I : \emptyset$ (пустое множество)
- $(\top)^I : \Delta$ (множество всех возможных объектов)
- $(C)^I : C \subseteq \Delta$ (множество объектов)
- $(\mathcal{R})^I : R \subseteq \Delta \times \Delta$ (бинарное отношение)
- $(C_1 \sqcap C_2)^I : C_1 \cap C_2$ (пересечение множеств)
- $(C_1 \sqcup C_2)^I : C_1 \cup C_2$ (объединение множеств)
- $(\neg C)^I := \Delta \setminus C$ (все объекты, не входящие в множество C)
- $(\forall \mathcal{R}. C)^I : \{p \mid \forall q . \langle p, q \rangle \in R \rightarrow q \in C\}$ (все значения роли R принадлежат множеству C)
- $(\exists \mathcal{R}. C)^I : \{p \mid \exists q . \langle p, q \rangle \in R \rightarrow q \in C\}$ (существует хотя бы одно значение отношения R , принадлежащее C)
- $(\leq_n \mathcal{R}. C)^I : \{p \mid \#\{q \mid \langle p, q \rangle \in R\} \geq n\}$ (множество значений отношения R , принадлежащих C , не меньше n)
- $(C_1 \sqsubseteq C_2)^I : C_1 \subseteq C_2$ (C_1 является подмножеством C_2)
- $(i:C)^I : I \in C$ (объект I принадлежит множеству C)



Примеры определений в логике описаний

- Примеры определений
 - $\text{Male} \sqsubseteq \text{Person}$
 - $\text{Female} \sqsubseteq \text{Person}$
 - $\text{Male} \sqsubseteq \neg \text{Female}$
 - $\text{Person} \sqsubseteq \text{hasFather.Male}$
 - $\text{Person} \sqsubseteq \text{hasMother.Female}$
 - $\text{hasMother} \sqsubseteq \text{hasParent}$
 - $\text{hasFather} \sqsubseteq \text{hasParent}$
 - $\text{Doctor} \sqsubseteq \text{Person}$
 - $\text{Woman} \equiv \text{Person} \sqcap \text{Female}$
 - $\text{Mother} \equiv \text{Woman} \sqcap \exists \text{hasChild.}\top$
 - $\text{Person} \sqsubseteq \forall \text{hasChild.Person}$
- Пример с индивидом
 - Mary – женщина, она не доктор, у нее есть дочка
 - $\text{Woman} \sqcap \neg \text{Doctor} \sqcap \exists \text{hasChild.Female}$



Задачи, решаемые в логиках описаний

- Задачи вывода в дескриптивных логиках
 - Доказательство выполнимости концепта C (satisfiability)
 - $(C)^I \neq \emptyset$, существует непустая интерпретация, соответствующая концепту C
 - Доказательство вложенности концептов $C_1 \sqsubseteq C_2$ (subsumption)
 - сводится к задаче выполнимости концепта $C_1 \sqcup \neg C_2$
 - Доказательство эквивалентности концептов
 - сводится к задаче обратной вложенности $C_1 \sqsubseteq C_2$ и $C_2 \sqsubseteq C_1$
 - Доказательство выполнимости теории
 - проверка всех концептов на выполнимость
 - Классификация концептов
 - вывод всех возможных отношений вложенности и эквивалентности между концептами
 - Классификация экземпляра
 - Нахождение концепта для экземпляра
 - Нахождение отношения для экземпляра
 - Нахождение всех экземпляров концепта
 - Ответы на конъюнктивные запросы и т. д.



Выбор логики

- Разные дескриптивные логики имеют разные наборы разрешённых конструкций
- Выбор конструкций обусловлен сложностью логического вывода для некоторых задач в данной логике
- От набора конструкций зависит сложность задач вывода
- В определённых дескриптивных логиках разрешим автоматический вывод для определённого набора задач



Конструкции OWL DL

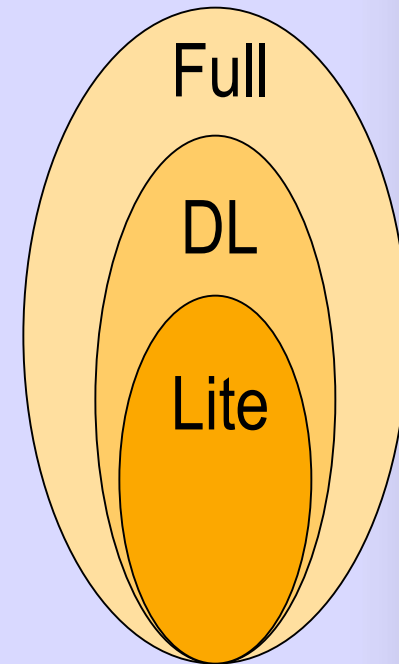
OWL DL	DL
Nothing	\perp
Thing	\top
Class	C
ObjectProperty	R
unionOf	$C_3 := C_1 \sqcup C_2$
complementOf	$C_2 := \neg C_1$
oneOf	$C := \{I_1\} \sqcup \{I_2\} \sqcup \dots$
allValuesFrom	$C_2 := \forall R.C_1$
someValuesFrom	$C_2 := \exists R.C_1$
hasValue	$C := \exists R.\{I\}$
minCardinality	$C := \geq_n R$
maxCardinality	$C := \leq_n R$
cardinality	$C := =_n R$
subClassOf	$C_1 \sqsubseteq C_2$
equivalentClass	$C_1 \equiv C_2 \equiv \dots$

OWL DL	DL
subPropertyOf	$R_1 \sqsubseteq R_2$
domain	$\leq_1 R \sqsubseteq C$
range	$\top \sqsubseteq \forall R.C$
inverseOf	$R_1 \equiv R_2^-$
equivalentProperty	$R_1 \equiv R_2 \equiv \dots$
TransitiveProperty	$R^+ \sqsubseteq R$
FunctionalProperty	$\top \sqsubseteq \leq_1 R$
InverseFunctional Property	$\top \sqsubseteq \leq_1 R^-$
SymmetricProperty	$R \equiv R^-$
DatatypeProperty	A
type	$I: C$
sameAs	$I_1 = I_2 = \dots$
differentFrom, AllDifferent	$I_1 \neq I_2 \neq \dots$
disjointWith	$C_1 \sqsubseteq \neg C_2$ $C_1 \sqcap C_2 \equiv \perp$



Профили OWL

- **OWL DL (Description Logic)**
 - Содержит все конструкции OWL, но ограничивает их интерпретацию и совместное использование для гарантированной вычислительной полноты и разрешимости, соответствует логике SHOIQ(D+)
 - Сложность задачи классификации: NExpTime
- **OWL Lite**
 - Подмножество OWL для выражения классификаций и простых отношений, соответствует логике SHIF(D)
 - Декларировалась возможность масштабирования по количеству определений
 - На практике сложность задачи классификации без считающих кванторов ExpTime (не сильно изменилась)
- **OWL Full**
 - Все конструкции OWL без ограничений использования, но без гарантий возможности вывода
 - Например, экземплярами понятий могут быть понятия
- **Совместимость определений**
 - Все корректные онтологии OWL Lite корректны для OWL DL и OWL Full.
 - Все корректные онтологии OWL DL корректны для OWL Full.





Профили OWL 2

- OWL 2 EL
 - Для большого количества классов
 - Логика EL конструкциями $\exists R.C$ и $C_1 \sqcap C_2$
 - и несколько конструкций, не ухудшающих вывод
- OWL 2 QL
 - Коррекции логики описаний для улучшения производительности
 - Некоторые ограничения стали неудобными, зависящими от контекста
- OWL 2 RL
 - приближенный к Datalog
 - продукции вида $R_1(\vec{x}_1) \wedge \dots \wedge R_n(\vec{x}_n) \rightarrow R(\vec{x})$



Пример OWL

```
Class (:Person
    SubClassOf(ObjectExactCardinality( 1 :hasParent :Man ))
    SubClassOf(ObjectExactCardinality( 1 :hasParent :Woman ))
)
SubObjectPropertyOf(:hasFather :hasParent)
InverseObjectProperties(:hasParent :hasChild)
DisjointClasses(:Woman :Man)
```



Примеры определений на языке OWL

- Классы персона, мужчина, женщина
 - Class (person)
 - Class (man
SubClassOf(person))
 - Class (woman
SubClassOf(person))
- Свойства быть родителем, матерью, отцом
 - ObjectProperty (hasParent)
 - ObjectProperty (hasMother
SubPropertyOf(hasParent))
 - ObjectProperty (hasFather
SubPropertyOf(hasParent))
- Каждая мать женщина
 - SubClassOf(
SomeValuesFrom(
InverseOf(hasMother) owl:Thing)
woman)
- Классы женщин и мужчин не пересекаются
 - DisjointClasses(man woman)
- У человека двое родителей
 - SubClassOf(person
ExactCardinality(2 hasParent))
- У человека есть мать и отец
 - SubClassOf(
person
IntersectionOf(

SomeValuesFrom(hasMother
owl:Thing)

SomeValuesFrom(hasFather
owl:Thing)))



Инструменты

- Редактор онтологий
 - Protégé <http://protege.stanford.edu/>
- Обертки
 - OWL API
 - Apache Jena (Java),
 - owlready (Python)
- Системы вывода
 - Racer
 - Pellet
 - FaCT
 - KAON2



Место онтологий в Семантическом вебе

- Человеко-читаемый означает, что смысл написанного несёт в себе человек, не компьютер
- Онтология даёт возможность выражения смысла непосредственно с помощью языка
- Онтологии в OWL определены в терминах языков RDF и RDFS (языков описания ресурсов)
- Онтологии в OWL являются основой для выражения правил в их терминах

