Lecture 1: Introduction to Knowledge Representation

2-AIN-144/2-IKV-131 Knowledge Reperesentation & Reasoning

Martin Baláž, Martin Homola

Department of Applied Informatics
Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava

14 Feb 2013

Course Info

Course:

- 2-AIN-144 Knowledge Representation & Reasoning
- 2-IKV-133 Representation, Knowledge & Reasoning

Lecturers:

- Martin Baláž, `balaz@ii.fmph.uniba.sk`
- Martin Homola, `teaching@tbc.sk`
- Alexander Šimko (labs)

Timetable:

- Lectures: Thursday 9:50 am room M-XII
- Labs: Tuesday 8:10 am room room M-II

Course homepage:

- `http://dai.fmph.uniba.sk/w/Course_KRR`

Evaluation

- Labs
- Project
- Midterm
- Exam

. . . details next week

Knowledge Representation & Reasoning

KR&R: part of AI concerned with representing knowledge with symbols and facilitating inference in deriving new knowledge from the existing knowledge.

Knowledge Representation & Reasoning

KR&R: part of AI concerned with representing knowledge with symbols and facilitating inference in deriving new knowledge from the existing knowledge.

KR&R: Applications

Applied on variety of problems:

- Database integrity checking
- Database integration
- Semantic interoperability of systems
- Inconsistence resolution
- Knowledge revision
- Reasoning about typical cases
- Planning
- Preference handling
- Generalization of knowledge (induction)
- Providing explanations, diagnosis (abduction)

Course Overview

- Introduction: from databases to KR&R
- From deduction to hypothetical reasoning
- Preferences
- Knowledge revision
- Induction
- Abduction

Relational Databases

Example relational database:

Student(name:str, grade:num):

name	grade
Jorge	1
Mike	7
Cecilia	3

Professor(name:str office:num):

name	office
Smiths	403
Jones	404

SupervisedBy(student:str, supervisor:str):

student	supervisor
Jorge	Mike
Mike	Smith
Cecilia	Jones

Transforming RDB into FOL Theory

- tables (n-ary relations) $\rightsquigarrow$ n-ary predicates
- values $\rightsquigarrow$ constants (nullary terms)
- table rows (records) $\rightsquigarrow$ ground atoms (facts)

Transforming RDB into FOL Theory

Example DB transforms into FOL theory $KB =$

{Student(Jorge, 1),
Student(Mike,7),
Student(Cecilia,3),
Professor(Smiths,403),
Professor(Jones,404),
SupervisedBy(Jorge, Mike),
SupervisedBy(Mike,Smith),
SupervisedBy(Cecilia, Jones)}

Querying Databases with FOL

Definition (Conjunctive query)

A conjunctive query (CQ) is a FOL formula $q(\vec{v})$ built on top of constants and variables (i.e., without any functional symbols), that has the form of a conjunction of atoms with free variables from $\vec{v}$, additional existential variables from $\vec{e}$.

Examples:

Querying Databases with FOL

Definition (Union of conjunctive queries)

A union of conjunctive queries (UCQ) is a set of CQ with (the same name and) the same set of free variables $\vec{v}$).

Examples:

Querying Databases with FOL

Definition (Answer to CQ, UCQ)

An answer to CQ (UCQ) $q(\vec{v})$ w.r.t. a FOL theory T is a set of ground atoms

$$\{q(\vec{v}/\theta) \mid T \models q(\vec{v}/\theta) \text{ for some ground substitution } \theta \text{ of } \vec{v}\}$$

Examples:

Querying Databases with FOL

Definition (Datalog)

A datalog program is logic (prolog) program consisting of rules of the form $A \leftarrow L_1, \ldots, L_n$ s.t.:

1. A is a function-free FOL atom, $L_1, \ldots, L_n$ are function free FOL literals
2. each variable occurring in A occurs also in some positive literal L_{i_k}, $1 \leq i_k \leq n$; and each variable occurring in a negative literal L_{i_j} occurs also in some positive literal L_{i_k}, $1 \leq i_j, i_k \leq n$

Definition (Query to datalog program)

A query (goal) to a datalog program is any function free atom $q(\vec{v})$ with free variables $\vec{v}$.

Querying Databases with FOL

Definition (Answer to a query w.r.t. datalog porogram)

An answer to a query $q(\vec{v})$ w.r.t. a datalog program P is a set of ground atoms

$$\{q(\vec{v}/\theta) \mid P \models q(\vec{v}/\theta) \text{ for some ground substitution } \theta \text{ of } \vec{v}\}$$

Examples: