Automatic Reasoner Selection using Machine Learning

International Conference on Web Intelligence, Mining and Semantics (WIMS '12)

Jürgen Bock, Uta Lösch, Hai Wang

June 13, 2012
Motivation

• Added value in the *Semantic Web*
  • Ability to automatically infer implicit knowledge from explicitly given axioms and facts
    → *Reasoning*
• Required: **ontologies** with logical underpinning
• Reasoning is complex
• Different reasoning scenarios (ontologies, queries)
• Plethora of reasoning systems with different strengths and weaknesses
• Selecting most suitable reasoner is difficult
• **Machine learning** classifiers can help
Ontology Reasoning

Reasoning:
Obtain implicit knowledge from explicitly given axioms and facts.

- Web Ontology Language (OWL) based on Description Logics
  - Entities
    (classes, individuals, object properties, data properties)
  - Axioms (statements about entity expressions)
    (subsumption, assertion, domain / range, ...)
- Reasoning tasks:
  class satisfiability, subsumption checking, instance retrieval, ...
- Reasoning algorithms:
  tableau, hypertableau, automata-based, resolution, ...
- Plethora of reasoning systems
- Different strengths and weaknesses
Ontology Reasoning
Reasoning Brokerage

Ontology Reasoning: Reasoning Brokerage

Diagram:
- HERAKLES
  - OWL API
  - Load Strategy
  - Execution Strategy
  - OWLLink
  - Semantic Application
    - OWLLink
  - Reasoner 1
  - Reasoner 2
  - Reasoner ...
  - Reasoner n
Classification Methods

Classification task:

Find function $f$ based on a set of training data $X$ consisting of instances $x$ for which the class $C_i \in \{C_1, \ldots, C_m\}$ is known, such that $f(x) = C_i$ for most training examples.

Reasoning request: (Ontology + Query)

Reasoners to select from

- Naïve Bayes (NB)
- K-Nearest Neighbour (KNN)
- Decision Tree (DT)
- Support Vector Machine (SVM)
Feature Definition
Ontology and Query Features

• Ontology Features
  • Description Logic expressiveness
  • Entity counts (different types)
  • Axiom counts (different types)

• Query features
  • Name (OWL API)
  • Return type
  • Parameter count
  • Parameter types
  • Axiom / class expression type
Implementation
HERAKLES Broker Strategy

MachLrnLStrategy
- Ontology feature analysis

MachLrnSPStrategy
- Query feature analysis
- Reasoner prediction using Weka
Evaluation
Setup / Training Data

• Data Set
  • 499 ontologies found on the Web
  • 36 query / ontology
    → 3175 reasoning requests (training instances) processable

• Reasoners
  • FaCT++: (2142 instances)
  • HermiT: (473 instances)
  • Pellet: (478 instances)
  • none: (82 instances)
Evaluation

Results

- 10-fold cross validation
- 3 of 4 classification methods outperform base line
- Best performance: DT (KNN and SVM competitive)
- NB: requires independent features (not guaranteed here, prior feature selection could improve results)
- Feature importance:
  - Query features more important than ontology features (esp. axiom type, class expression type, query name)
  - Annotation property count relevant
  - Description Logic expressiveness less relevant

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base line (all FaCT++)</td>
<td>0.615</td>
<td>0.763</td>
<td>0.615</td>
<td>0.681</td>
</tr>
<tr>
<td>Naive Bayes</td>
<td>0.548</td>
<td>0.628</td>
<td>0.548</td>
<td>0.545</td>
</tr>
<tr>
<td>$k$-nearest neighbours ($k=20$, weight by $1 - distance$)</td>
<td>0.728</td>
<td>0.709</td>
<td>0.728</td>
<td>0.710</td>
</tr>
<tr>
<td>Decision Trees ($J48$, $minNumObj = 10$, $confidenceFactor = 0.5$)</td>
<td><strong>0.774</strong></td>
<td><strong>0.756</strong></td>
<td><strong>0.774</strong></td>
<td><strong>0.760</strong></td>
</tr>
<tr>
<td>Support Vector Machine ($\gamma = 0.5$)</td>
<td>0.708</td>
<td>0.686</td>
<td>0.708</td>
<td>0.665</td>
</tr>
</tbody>
</table>

13/06/2012
Summary / Outlook

- Automatic reasoning as added value of using Semantic Technologies
- Available reasoning systems demonstrate different performance for different reasoning requests
- Machine learning applied to select most suitable reasoner for any given request
- Implementation as HERAKLES broker strategy
- Prediction accuracy of up to 77% (decision tree) → 16% increase compared to base line

- Future work
  - Dynamic model adjustment
  - OWL 2 profile reasoners
  - Fine-grained feature definitions (combinations of ontology language features)
Thanks for your attention!

http://herakles.sourceforge.net
Feature Definition
Ontology Features

- **Description Logic expressiveness**
  - Nominal values (ALC, SHIQ, SHOIQ, SROIQ, ...)

- **Counts**
  - **Referenced entities**
    (classes, object properties, data properties, individuals)
  - **Class axioms**
    (subsumption, equivalence, disjointness, GCIs)
  - **Object property axioms**
    (subsumption, equivalence, disjointness, functional, inverse, transitive, ..., domain, range, subproperty chain)
  - **Data property axioms**
    (subsumption, equivalence, disjointness, functional, domain, range)
  - **Individual axioms**
    (class / (neg.) object property / (neg.) data property assertion, same, different)
  - **Annotation axioms**
    (assertion, domain, range)
  - **Others**
    (declaration axioms)
Feature Definition
Query Features

- Typical reasoning tasks (subsumption checking, class satisfiability, instance retrieval, ...)
  - Can all be reduced to satisfiability checking

- Reasoner optimisations on a higher level
  - OWL API reasoning (convenience) methods
    - Get unsatisfiable classes
    - Get object property values
    - ...

- Query features
  - Name (OWL API)
  - Return type
  - Parameter count
  - 1\textsuperscript{st} parameter type
  - 2\textsuperscript{nd} parameter type
  - Axiom type
  - Class expression type
  - Depth of class expression
  - Atomic class count in class expression
Implementation
Training Data Generator

Ontologies → Ontology Loader → Query Generator → Reasoner Tester → Weka

Ontology Feature Extractor → Query Feature Extractor → Query Data Aggregator
Class Expression Example

ObjectUnionOf

ObjectSomeValuesFrom

ObjectProperty

Class

DataSomeValuesFrom

int

DataRange

Class

Class