Efficient Instance Retrieval over Semi-Expressive Ontologies

Dissertation Presentation

Sebastian Wandelt
Hamburg, 6th of October 2011

Chairman: Professor Volker Turau (Hamburg University of Technology)
Reviewer: Professor Ralf Möller (Hamburg University of Technology)
Reviewer: Professor Ian Horrocks (University of Oxford)
Reviewer: Professor Norbert Ritter (University of Hamburg)
Overview

• Motivation
• Research Objectives and Methodology
• Main Contributions and Evaluation
• Discussion and Future Work
Motivation and Background

- Semantic Web
- Ontologies / Description logics
- Reasoning is hard (expressivity vs. scalability)
- Thesis:
  
  “Instance retrieval for the description logic SHI”
Related Work

• Less expressive description logics
  – DL-Lite [ACKZ09]

• Sound only
  – Triple Stores [AG10], Approximations [RPZ10]

• Sound and complete
  – SHER [DFK09]
  – GCQs/CQs [HM08], [SBPKK07]
  – Rewriting [HMS07, HKRT08], Hypertableau [MSH09]
  – Instance Store [SHT05]

• Neither sound, nor complete
  – Approximations [TGH10]
Research Objectives

• Release the main-memory dependency from DL reasoning systems
• Focus on
  – Semi-expressive DLs (SHI), no datatypes
  – Large ABox, mid-size TBox / RBox
  – Atomic instance retrieval queries
• Prepare for ontology changes
Scientific Methodology

- Practical work
- Well-documented implementation
- Proofs
- Runs on off-the-shelf laptop
  - Intel C3 2.4 GHz, 4 GB RAM, 500GB, Windows 7, Java 6
- Evaluation with benchmark ontology

=> Reproducibility and repeatability
In the following ...

- Instance checking
- Instance retrieval
- Ontology changes
Instance Checking

\[ T = \{\text{Chair} \equiv \exists \text{headOf}.\text{Department}, \text{Student} \equiv \exists \text{takes}.\text{Course}, \]
\[ \text{UndergraduateCourse} \sqsubseteq \text{Course}, \text{Course} \cap \text{Chair} \sqsubseteq \bot, \top \sqsubseteq \forall \text{takes}.\text{Course}, \]
\[ \top \sqsubseteq \forall \text{teaches}.\text{Course}, \exists \text{memberOf}.\top \sqsubseteq \text{Professor}\} \]
\[ \mathcal{R} = \{\text{headOf} \sqsubseteq \text{memberOf}, \text{teaches} \equiv \text{isTaughtBy} -, \text{Trans} (\text{suborgOf})\} \]
\[ \mathcal{A} = \{\text{Department} (\text{cs}), \text{Department} (\text{ee}), \]
\[ \\text{Professor} (\text{ann}), \text{Professor} (\text{eve}), \text{Professor} (\text{mae}), \]
\[ \text{UndergraduateCourse} (\text{c1}), \text{UndergraduateCourse} (\text{c4}), \]
\[ \text{UndergraduateCourse} (\text{c5}), \]
\[ \text{GraduateCourse} (\text{c2}), \text{GraduateCourse} (\text{c3}), \text{Student} (\text{ani}), \text{Student} (\text{ean}), \]
\[ \text{Student} (\text{eva}), \text{Student} (\text{noa}), \text{Student} (\text{sam}), \text{Student} (\text{sue}), \text{Student} (\text{zoe}), \]
\[ \text{headOf} (\text{ann}, \text{cs}), \text{memberOf} (\text{eve}, \text{cs}), \text{headOf} (\text{mae}, \text{ee}), \]
\[ \text{teaches} (\text{ann}, \text{c1}), \text{teaches} (\text{eve}, \text{c2}), \text{teaches} (\text{eve}, \text{c3}), \]
\[ \text{teaches} (\text{mae}, \text{c4}), \text{teaches} (\text{mae}, \text{c5}), \]
\[ \text{takes} (\text{ani}, \text{c1}), \text{takes} (\text{ean}, \text{c1}), \text{takes} (\text{ean}, \text{c2}), \text{takes} (\text{eva}, \text{c3}), \]
\[ \text{takes} (\text{noa}, \text{c3}), \text{takes} (\text{sam}, \text{c4}), \text{takes} (\text{sue}, \text{c5}), \text{takes} (\text{zoe}, \text{c5})\} \]
ABox Split

• Break up a role assertion:
ABox Split – Active Students?

\{Professor \subseteq \forall \text{teaches}.\text{Course}, \text{Course} \subseteq \forall \text{hasStudent}.\text{ActiveStudent}\}

Diagram:

- Professor
  - eve
  - teaches
  - c3
  - hasStudent
  - noa

ABox splits:

- Professor
  - eve
  - teaches
  - c3
  - hasStudent
  - noa

- Professor
  - eve
  - teaches
  - c3
  - hasStudent
  - noa

- Professor
  - noa
  - hasStudent
  - c3
ABox Split – Active Students?

\{ Professor \sqsubseteq \forall \text{teaches}.Course, \text{Course} \sqsubseteq \forall \text{hasStudent}.\text{ActiveStudent} \}\cup \{\text{GradCourse} \sqsubseteq \text{Course}\}

Diagram:

- Professor \sqsupseteq \forall \text{teaches}.Course, course \sqsubseteq \forall \text{hasStudent}.\text{ActiveStudent}
- GradCourse \sqsubseteq \text{Course}

Diagram showing relationships between Professor, GradCourse, and their respective classes and students.
Definition (\(SHI\)-splittability of Role Assertions):
Given a \(SHI\)-ontology \(\mathcal{O} = \langle T, R, A \rangle\) and a role assertion \(R(a, b)\), we say that \(R(a, b)\) is \(SHI\)-splittable with respect to \(\mathcal{O}\) if

1. there exists no transitive role \(R_2\) with respect to \(\mathcal{O}\), such that \(\mathcal{O} \models R \sqsubseteq R_2\),
2. for each \(C \in extinfo_{T,R}^\forall (R)\)
   - \(C = \bot\) or
   - there exists a concept description \(C_2\), such that \(C_2(b) \in A\) and \(T \models C_2 \sqsubseteq C\) or
   - there exists a concept description \(C_2\), such that \(C_2(b) \in A\) and \(T \models C \cap C_2 \sqsubseteq \bot\)
3. for each \(C \in extinfo_{T,R}^\forall (R^-)\)
   - \(C = \bot\) or
   - there exists a concept description \(C_2\), such that \(C_2(a) \in A\) and \(T \models C_2 \sqsubseteq C\) or
   - there exists a concept description \(C_2\), such that \(C_2(a) \in A\) and \(T \models C \cap C_2 \sqsubseteq \bot\).
Instance Checking: Individual Islands
Instance Checking: Individual Islands

- Small modules fitting into main memory
- Note: we do not have to perform ABox splits in practice!

![Diagram showing relationships between concepts and individuals, including labels like ee, headOf, mae, teaches, c4, c5, mae, teaches, c5, takes, sue, zoe.](image)
Instance Checking: Evaluation

- Graph 1: Number of individuals vs. number of universities
- Graph 2: Role assertion unsatisfiability vs. number of universities
- Graph 3: Average module size vs. number of universities
Instance Retrieval: Similarity

- Many (small) islands are similar to each other
- => use of homomorphisms

- Example: 9 instead of 17 instance checks
Instance Retrieval: Evaluation

LUBM(10000) = 1.4 billion ABox assertions

IR: Chair? [SGH10]

IR: University?
Ontology Changes

- Syntactic ontology updates
  - Keep complex data structures updated

- Non-atomic queries
  - As long as the query-concept does not contain existential constraints - and does not change the role hierarchy, nothing has to be recomputed (individual islands would only become more small)!
  - In the other case, new role assertions can become unsplittable!
Ontology Changes: Evaluation

Graph 1: Load time in minutes vs. number of universities.
- Load time increases linearly with the number of universities.

Graph 2: Memory usage in MB vs. number of universities.
- Memory usage remains relatively stable across the range of universities.

Slide 20 / 24
Ontology Changes: TBox / RBox

• Hard to find representative update
  – From adding: \(\text{FullProfessor} \subseteq \text{Professor}\)
  – ... over removing: \(\text{Student} \equiv \exists \text{takes.Course}\)
  – ... to (high impact) RBox-updates
Analysis

• Pro:
  – Very good for ontologies with many (mainly) integrity constraints
  – ABox updates are *local* and usually *fast*

• Con:
  – *Computational* ontologies
  – Complex updates of the terminology can be slow
Conclusions / Scientific Contributions

• ABox modularization techniques
  Sebastian Wandelt, Ralf Möller: Island Reasoning for ALCHI Ontologies, FOIS 2008

• Optimized instance retrieval
  Sebastian Wandelt et.al.: Towards Scalable Instance Retrieval over Ontologies, J. of Software and Informatics 2010

• Parallelization of instance retrieval
  Sebastian Wandelt, Ralf Möller: Distributed Island-Based Query Answering for Expressive Ontologies, DL 2010

• Updating data structures under changes to the ontology
  Sebastian Wandelt, Ralf Möller: Updatable Island Reasoning for ALCHI-ontologies, KEOD 2009

• Instance retrieval can be solved for LUBM(10000)
  Sebastian Wandelt, Ralf Möller: Sound and Complete Instance Retrieval for 1 Billion ABox Assertions, SSWS 2011
Future Work

• Optimization of retrieval process from the database
• More expressive query languages
• More expressive ontology languages
• Evaluation on more real world datasets

Not in competition with DL reasoner ... results help them to deal with large ontologies more efficiently!
References I

- [AG10]: AllegroGraph RDFStore Web 3.0's Database, http://www.franz.com/agraph/allegrograph/
References II

- **[RPZ10]**: Yuan Ren, Jeff Z. Pan and Yuting Zhao. Towards Soundness Preserving Approximation for ABox Reasoning of OWL2. The International Description Logic Workshop (DL2010). 2010
- **[SHT05]**: Sean Bechhofer, Ian Horrocks, and Daniele Turi. The OWL Instance Store: System Description. In Proc. of the 20th Int. Conf. on Automated Deduction (CADE-20), Lecture Notes in Artificial Intelligence, pages 177-181. Springer, 2005