Fast Relational Learning Using Bounded LGG

Author: Andrea Fuksová[1], Advisor: Ondřej Kuželka[2]

[1]Czech Technical University in Prague, [2]KU Leuven

Problem

Sometimes, data expressed by relations make better sense than expressed as vectors of real numbers. **Relational machine learning:**

- Subfield of machine learning
- Learning from structured data
- Structures encoded as:
  - Labelled graphs
  - First order logic clauses
  - Relational structures
- So far, most theory based on first order logic formulation (FOL)

Preliminaries

Def. 1. ***Vocabulary*** *σ is a finite set of relation symbols with associated an arity.*

Def. 2. ***Relational structure*** *$\mathbb{A}$ of type σ is a pair of universe $\mathcal{U}_A$ and a sequence of relations $\mathcal{R}_A$. There exists one relation $R^A \in \mathcal{R}_A$ for each $R \in \sigma$ with the same arity as R.*

Def. 3. *A* ***homomorphism*** *from a structure $\mathbb{A}$ to a structure $\mathbb{B}$ of the same type is a mapping $f : \mathcal{U}_A \to \mathcal{U}_B$ such that for every $m-ary$ $R \in \sigma$ and every $(a_1, \ldots, a_m) \in R^A$ we have $(f(a_1), \ldots, f(a_m)) \in R^B$. If this homomorphism exists, we denote it by $\mathbb{A} \to \mathbb{B}$. If $\mathbb{A} \to \mathbb{B}$ and $\mathbb{B} \to \mathbb{A}$ we say that $\mathbb{A}$ and $\mathbb{B}$ are* ***homomorphically equivalent*** *(denoted by $\mathbb{A} \approx \mathbb{B}$).*

Def. 4. *A relational structure $\mathbb{C}$ is said to be a* ***least general generalization (LGG)*** *of the relational structures $\mathbb{A}$ and $\mathbb{B}$ if and only if $\mathbb{C} \to \mathbb{A}$ and $\mathbb{C} \to \mathbb{B}$ and for every other relational structure $\mathbb{D}$ such that $\mathbb{D} \to \mathbb{A}$ and $\mathbb{D} \to \mathbb{B}$ it holds $\mathbb{D} \to \mathbb{C}$.*

Goal

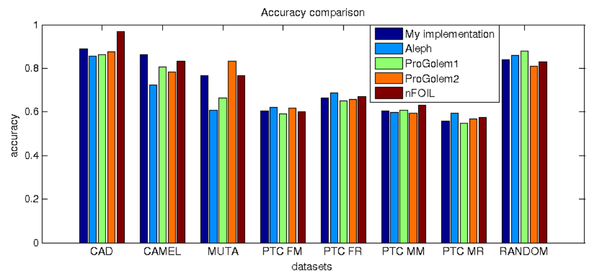
- Input: sets E^+ and E^- of positive and negative examples
- Examples are relational structures
- Find a classifier: set S of relational structures
- Structure e classified as:
  - positive $\Leftrightarrow \exists s \in S : s \to e$
  - negative otherwise
- If $s \to e$, we say that s **covers** e

Principle

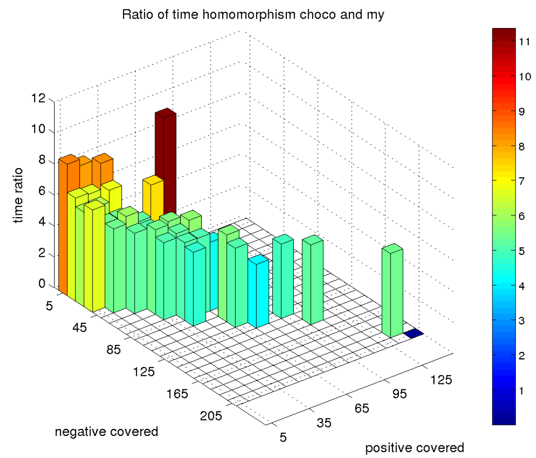
- Learning based on application of LGG on positive examples
- Homomorphism can be formulated as a **Constraint satisfaction problem (CSP)**
- Deciding about homomorphism for two structures is NP-complete
- Basic algorithm for finding LGG produces very large structures which need to be reduced
- Reduction without generality loss: find smallest homomorphically equivalent structure
- Result: Basic learning requires a lot of computationally costly homomorphism tests
- Idea: Exploiting polynomial-time local consistency techniques from CSP to test so called **bounded homomorphism**

Some results

- Effective implementation of in general exponential-time methods based on complete CSP solution is usually faster than solution based on polynomial-time bounded operations (exploiting local consistency techniques)



- Results **comparable in accuracy with state-of-the-art algorithms** for relational machine learning
- Figure shows accuracy performance of my implementation and state-of-the-arts algorithms on eight datasets.
- Every algorithm has its own color



- My CSP solver performs on our tasks **faster than widely used Choco CSP solver**
- Figure shows ratio of average runtime of homomorphism test using Choco CSP solver / homomorphism test using my CSP solver
- Measured average runtime of homomorphism testing of a random structure to all structures in a data set
- Dependence on number of positive and negative examples covered

My work

- Reformulation of theory from FOL into terms of relational structures.
- This formulation should be **more accessible** for most scientific audience as opposed to FOL
- **Effective and complex** implementation of the studied methods in Java
- Implementation of a **new effective CSP solver**
- Investigation of runtime and accuracy performance of the methods

Example

- Results on dataset containing 80 Hexose-binding **protein domains** (positive examples) and 80 non-Hexose-binding protein domains (negative examples).
- Presented at the workshop Machine Learning in Computational Biology at the conference **NIPS 2013**
- Equivalent encoding as labelled graphs
- One vertex for every atom (labelled by the atom type + position in the amino acid)
- Edge labelled by a discretized distance (if < 4 Angstroms).
- 10-fold cross-validation accuracy 71.9 ± 5.3
- Picture: structure covering covers 39 positive examples and no negative example