Machine Learning for the Semantic Web: filling the gaps in Ontology Mining

Claudia d’Amato

Department of Computer Science
University of Bari

EGC 2017 ◊ Grenoble, January 26, 2017
**Semantic Web goal:** making data on the Web machine understandable

- ontologies act as a *shared vocabulary for assigning data* semantics

Examples of existing real ontologies

- Schema.org
- Gene Ontology
- Foundational Model of Anatomy ontology
- Financial Industry Business Ontology (by OMG Finance Domain Task Force)
- GoodRelations
- ...
Reasoning on Description Logics Ontologies

OWL adopted $\Rightarrow$ **Description Logics** theoretical foundation
OWL adopted ⇒ Description Logics theoretical foundation

Ontologies are equipped with deductive reasoning capabilities ⇒ allowing to make explicit, knowledge that is implicit within them

Deduction:
”Crédit du Nord”, ”Crédit Agricole” are also Company
Introduction & Motivation

Reasoning on Description Logics Ontologies

Question: would it be possible to discover new/additional knowledge by exploiting the evidence coming from the assertional data?

Deduction: "Crédit du Nord", "Crédit Agricole" are also Company.

"UniCredit" is a Bank.

Incompleteness
Reasoning on Description Logics Ontologies

Inconsistency

Mellon cannot be a Person and a Bank
Introduction & Motivation

Reasoning on Description Logics Ontologies

Question: would it be possible to discover new/additional knowledge by exploiting the evidence coming from the assertional data?

Deduction: "Crédit du Nord", "Crédit Agricole" are also Company

Idea: exploiting Machine Learning methods for Ontology Mining related tasks

[d'Amato et al. @SWJ'10]
Reasoning on Description Logics Ontologies

**Question:** would it be possible to discover new/additional knowledge by exploiting *the evidence coming from the assertional data*?

---

**Diagram:***

- **T-Box:**
  - Person
  - Bank
  - subclassOf
- **A-Box:**
  - John
  - givesCredit
  - hasType
  - Noise:
    - Person $\equiv \neg$ Bank
    - missing
Introduction & Motivation

Reasoning on Description Logics Ontologies

**Question:** would it be possible to discover new/additional knowledge by exploiting *the evidence coming from the assertional data*?

![Ontology Diagram]

**Idea:** exploiting **Machine Learning methods** for **Ontology Mining** related tasks

* [d’Amato et al. @SWJ’10]
Definition (Ontology Mining)

All activities that allow for

**discovering hidden knowledge** from
ontological knowledge bases
Definition (Ontology Mining)

All activities that allow for discovering hidden knowledge from ontological knowledge bases.

Machine Learning (ML) methods

- focus on the development of methods and algorithms that can teach themselves to grow and change when exposed to new data

Special Focus on:

- (similarity-based) *inductive learning methods*
  - use specific examples to reach general conclusions
  - are known to be very efficient and fault-tolerant
Induction vs. Deduction

Deduction (Truth preserving)
Given:
• a set of general axioms
• a proof procedure
Draw:
• correct and certain conclusions

Induction (Falsity preserving)
Given:
• a set of examples
Determine:
• a possible/plausible generalization covering
  • the given examples/observations
  • new and not previously observed examples
Ontology Mining Tasks

- Instance Retrieval (Instance Level)
- Concept Drift and Novelty Detection (Ontology Dynamic)
- Ontology Enrichment (Schema/Instance Level)

from an inductive perspective

Focus on: similarity-based methods
Ontology Mining Tasks

- **Instance Retrieval** (Instance Level)
- Concept Drift and Novelty Detection (Ontology Dynamic)
- Ontology Enrichment (Schema/Instance Level)

*from an inductive perspective*
Introducing Instance Retrieval

*Instance Retrieval* → Finding the extension of a query concept

- Instance Retrieval (Bank) = {"Crédit du Nord", "Crédit Agricole"}
**Problem:** Instance Retrieval in **incomplete/inconsistent/noisy** ontologies
Problem: Instance Retrieval in incomplete/inconsistent/noisy ontologies
Problem: Instance Retrieval in incomplete/inconsistent/noisy ontologies
Issues & Solutions I

IDEA

**Casting** the problem as a Machine Learning *classification problem*

*assess the class membership of individuals in a Description Logic (DL) KB w.r.t. the query concept*

**State of art classification methods cannot be** straightforwardly applied

- generally applied to *feature vector* representation
  \[\rightarrow \text{upgrade DL expressive representations}\]
- implicit *Closed World Assumption* made in ML
  \[\rightarrow \text{cope with the Open World Assumption made in DLs}\]
- classes considered as *disjoint*
  \[\rightarrow \text{cannot assume disjointness of all concepts}\]
Adopted Solutions:

- Defined new semantic similarity measures for DL representations
  - to cope with the high expressive power of DLs
  - to deal with the semantics of the compared objects (concepts, individuals, ontologies)
  - to convey the underlying semantics of KB
- Formalized a set of criteria that a similarity function has to satisfy in order to be defined *semantic* [d’Amato et al. @ EKAW 2008]
- Definition of the classification problem taking into account the OWA
- Multi-class classification problem decomposed into a set a smaller classification problems
Definition (Problem Definition)

Given:
- a populated ontological knowledge base \( KB = (T, A) \)
- a query concept \( Q \)
- a training set with \{+1, -1, 0\} as target values

Learn a classification function \( f \) such that: \( \forall a \in \text{Ind}(A) : \)
- \( f(a) = +1 \) if \( a \) is instance of \( Q \)
- \( f(a) = -1 \) if \( a \) is instance of \( \neg Q \)
- \( f(a) = 0 \) otherwise (unknown classification because of OWA)

Dual Problem
- given an individual \( a \in \text{Ind}(A) \), tell concepts \( C_1, \ldots, C_k \) in \( KB \) it belongs to
- the multi-class classification problem is decomposed into a set of ternary classification problems (one per target concept)
Developed methods

Pioneering the Problem
- relational K-NN for DL KBs \[d’Amato et al. ESWC’08\]

Improving the efficiency
- kernel functions for kernel methods to be applied to DLs KBs \[Fanizzi, d’Amato et al. @ ISMIS’06, JWS 2012; Bloehdorn and Sure @ ISWC’07\]

Scaling on large datasets
- Statistical Relational Learning methods for large scale and data sparseness \[Huang et al. @ ILP’10, Minervini et a. @ ICMLA’15\]
Example: Nearest Neighbor Classification

query concept: **Bank** \( k = 7 \)

target values standing for the class values: \( \{+1, 0, -1\} \)

class\( (x_q) \leftarrow ? \)
Example: Nearest Neighbor Classification

query concept: Bank \( k = 7 \)
target values standing for the class values: \( \{+1, 0, -1\} \)

\[
\text{class}(x_q) \leftarrow +1
\]
On evaluating the Classifier

**Problem:** How evaluating classification results?

- **Inductive Classification compared with a standard reasoner** *(Pellet)*
- Query concepts from ontologies publicly available considered
- Registered *mismatches*: **Induction**: \{+1, −1\} - **Deduction**: no results
- **Evaluated as mistake if precision and recall were used** while it could turn out to be a correct inference when judged by a human

**Defined new metrics to distinguish induced assertions from mistakes**

<table>
<thead>
<tr>
<th></th>
<th>Reasoner</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+1</td>
</tr>
<tr>
<td><strong>Inductive</strong></td>
<td><strong>M</strong></td>
</tr>
<tr>
<td><strong>Classifier</strong></td>
<td></td>
</tr>
<tr>
<td><strong>+1</strong></td>
<td><strong>M</strong></td>
</tr>
<tr>
<td><strong>0</strong></td>
<td><strong>O</strong></td>
</tr>
<tr>
<td><strong>-1</strong></td>
<td><strong>C</strong></td>
</tr>
</tbody>
</table>

*M* Match Rate  
*C* Commission Error Rate  
*O* Ommission Error Rate  
/ Induction Rate
Lesson Learnt from experiments I

- **Commission error** almost zero on average
- **Omission error rate** very low and only in some cases
  - Not null for ontologies in which disjoint axioms are missing
- **Induction Rate** not zero
  - new knowledge (not logically derivable) induced ⇒ can be used for semi-automatizing the ontology population task
  - induced knowledge ⇒ individuals are instances of many concepts and homogeneously spread w.r.t. the several concepts.

<table>
<thead>
<tr>
<th></th>
<th>match</th>
<th>commission</th>
<th>omission</th>
<th>induction</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWM</td>
<td>97.5 ± 3.2</td>
<td>0.0 ± 0.0</td>
<td>2.2 ± 3.1</td>
<td>0.3 ± 1.2</td>
</tr>
<tr>
<td>LUBM</td>
<td>99.5 ± 0.7</td>
<td>0.0 ± 0.0</td>
<td>0.5 ± 0.7</td>
<td>0.0 ± 0.0</td>
</tr>
<tr>
<td>NTN</td>
<td>97.5 ± 1.9</td>
<td>0.6 ± 0.7</td>
<td>1.3 ± 1.4</td>
<td>0.6 ± 1.7</td>
</tr>
<tr>
<td>Financial</td>
<td>99.7 ± 0.2</td>
<td>0.0 ± 0.0</td>
<td>0.0 ± 0.0</td>
<td>0.2 ± 0.2</td>
</tr>
</tbody>
</table>
Ontology Mining Tasks

- Instance Retrieval (Instance Level)
- **Concept Drift and Novelty Detection** (Ontology Dynamic)
- Ontology Enrichment (Schema/Instance Level)

from an inductive perspective
Concept Drift and Novelty Detection

- Ontologies evolve over the time ⇒ *New assertions* added.

- **Concept Drift**
  - change of a concept towards a more general/specific one w.r.t. the evidence provided by new annotated individuals
    - almost all *Worker* work for more than 10 hours per days ⇒ *HardWorker*

- **Novelty Detection**
  - isolated cluster in the search space that requires to be defined through new emerging concepts to be added to the KB
    - subset of *Worker* *employed* in a company ⇒ *Employee*
    - subset of *Worker* *working for* several companies ⇒ *Free-lance*
Ontologies evolve over the time ⇒ *New assertions* added.

**Concept Drift**
- change of a concept towards a more general/specific one w.r.t. the evidence provided by new annotated individuals

**Novelty Detection**
- isolated cluster in the search space that requires to be defined through new emerging concepts to be added to the KB

**Question:** would it be possible to *automatically capture* them by analyzing the data configuration/distribution?

**IDEA**
Exploiting *(Conceptual) clustering methods* for the purpose
**Basics on Clustering Methods**

**Clustering methods**: unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

- intra-cluster *similarity* is high
- inter-cluster *similarity* is low
Basics on Clustering Methods

**Clustering methods:** unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

- intra-cluster *similarity* is high
- inter-cluster *similarity* is low
Basics on Clustering Methods

**Clustering methods**: unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

- intra-cluster *similarity* is high
- inter-cluster *similarity* is low
Clustering Individuals of An Ontology: Developed Methods

Purely Logic-based
- KLUSTER [Kietz & Morik, 94]
- CSKA [Fanizzi et al., 04]
  - Produce a flat output
  - Suffer from noise in the data

Similarity-based ⇒ noise tolerant
- Evolutionary Clustering Algorithm around Medoids [Fanizzi et al. @ IJSWIS 2008]
  - automatically assess the best number of clusters
- k-Medoid (hierarchical and fuzzy) clustering algorithm [Fanizzi et al. @ ESWC’08, Fundam. Inform.’10]
  - number of clusters required
- Terminological Cluster Trees [Rizzo et al. @ URSW’16]
  - extension of terminological decision trees
  - automatic number of clusters
Automated Concept Drift and Novelty Detection 1/3

Global Decision Boundary

C. d’Amato (UniBa)
The new instances are considered to be a *candidate* cluster.

An *evaluation* of it is performed for assessing its nature.
The new instances are considered to be a *candidate* cluster.
Automated Concept Drift and Novelty Detection 3/3

Candidate Cluster

Drift

Global Decision Boundary

Novelty
Lesson Learnt from Experiments

Clustering algorithms
- applied on ontologies publicly available
- *evaluated by the use of standard validity clustering indexes* (e.g. Generalized Dunns index, cohesion index, Silhouette index)

**Necessity of a domain expert/gold standard** particularly for validating the concept novelty/drift
Ontology Mining Tasks

- Instance Retrieval (Instance Level)
- Concept Drift and Novelty Detection (Ontology Dynamic)
- **Ontology Enrichment** *(Schema/Instance Level)*

*from an inductive perspective*
Ontology enrichment as a Concept Learning Problem
**On Learning Concept Descriptions I**

- Discovered clusters are only extensionally defined
- Having an intensional description for them could allow to **enrich the ontology at terminological level**

**Question:** How to learn concept descriptions automatically, given a set of individuals?

**IDEA**

Regarding the problem as a *supervised concept learning* task

**Supervised Concept Learning:**

- Given a training set of **positive** and **negative** examples for a **concept**,
- *construct a description* that will accurately classify whether future examples are positive or negative.
Definition (Problem Definition)

- **Given**
  - the KB $\mathcal{K}$ as a background knowledge
  - individuals in a cluster $C$ as positive examples
  - the individuals in the other clusters as negative examples

- **Learn**
  - a DL concept description $D$ so that
  - the individuals in the target cluster $C$ are instances of $D$ while those in the other clusters are not
Developed Methods for Supervised Concept Learning

For DLs that allow for (approximations of) the $msc$ and $lcs$, (e.g. \textit{ALC} or \textit{ALE}): 

- given a cluster $C_j$, 
  - $\forall a_i \in C_j$ compute $M_i := msc(a_i)$ w.r.t. the ABox $\mathcal{A}$ 
  - let $MSCs_j := \{M_i | a_i \in \text{node}_j\}$ 

- $C_j$ intensional description $lcs(MSCs_j)$

**Separate-and-conquer approach**

- YinYang [Iannone et al. @ Appl. Intell. J. 2007]
- DL-FOIL [Fanizzi et al. @ ILP 2008]
- DL-Learner [Lehmann et al. @ MLJ 2010, SWJ 2011]

**Divide-and-conquer approach**

- TermiTIS [Fanizzi et al. @ ECML 2010, Rizzo et al. @ ESWC 2015]
Separate and Conquer: Example

\[ C_1 = \text{MasterStudent} \quad C_1' = \text{MasterStudent} \sqcap \exists \text{workIn.}\top \]
\[ C_2 = \text{BachelorStudent} \quad C_2' = \text{BachelorStudent} \sqcap \exists \text{workIn.}\top \]
Examples of Learned Concept Descriptions with DL-FOIL

**BioPax**

*induced:*
\[ \text{Or( And( physicalEntity protein) dataSource)} \]

*original:*
\[ \text{Or( And( And( dataSource externalReferenceUtilityClass) ForAll(ORGANISM ForAll(CONTROLLED physicalInteraction))) protein)} \]

**NTN**

*induced:*
\[ \text{Or( EvilSupernaturalBeing Not(God))} \]

*original:*
\[ \text{Not(God)} \]

**Financial**

*induced:*
\[ \text{Or( Not(Finished) NotPaidFinishedLoan Weekly)} \]

*original:*
\[ \text{Or( LoanPayment Not(NoProblemsFinishedLoan))} \]
Ontology enrichment as a Pattern Discovery Problem
Research Idea

**Idea:** exploiting the evidence coming from the assertional data for *discovering hidden knowledge patterns* to be used for

1. obtaining new/additional assertional knowledge
2. suggesting new knowledge axioms (schema level)
3. extending existing ontologies with rules
   
   *while maintaining the decidability of the reasoning operators*

**Research Direction:** discovering hidden knowledge patterns *in the form of relational association rules* (ARs) [*d’Amato et al. @ SAC 2016*]
Developed Methods

RDF data

- performing descriptive and predictive task
- **no background knowledge** and **reasoning capability** exploited [Völker & Niepert @ ESWC’11; Galárraga et al. @ WWW’13, VLDB J.’15]
- association rules exploited for performing RDF data compression [Joshi, Hitzler et al. @ ESWC 2013]

Hybrid source of Knowledge

- discovering frequent patterns from DB plus ontology [Lisi @ IJSWIS 7(3), 2011, Józefowska et al. @ TPLP 10(3), 2010, d’Amato et al. @ URSW (LNCS Vol.)’14]

Ontological Knowledge Bases (**focused**)

- performing descriptive and **predictive task**
- **background knowledge** and **reasoning capability** exploited [d’Amato et al. @ SAC’16, EKAW’16]
Definition (Problem Definition)

Given:

- a populated ontological knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$
- a minimum ”frequency threshold” (fr_thr)

Discover:

- all frequent hidden patterns, with respect to fr_thr, in the form of relational association rules that may induce new assertions for $\mathcal{K}$. 
Definition (Relational Association Rule)

Given

- a populated ontological knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$,

a relational association rule $r$ for $\mathcal{K}$ is a horn-like clause of kind

(body $\rightarrow$ head)

where:

- **body** represents an abstraction of a set of assertions in $\mathcal{K}$ co-occurring with respect to fr_thr
- **head** represents a possibly new assertion induced from $\mathcal{K}$ and **body**

SWRL [Horrocks et al.@ WWW’04] is adopted as representation language.

- allows to extends the OWL axioms of an ontology with Horn-like rules
- The result is a KB with an enriched expressive power.
Discovering SWRL rules of the form:

\[ C_1(x) \land R_1(x, y) \land \cdots \land C_n(z) \land R_l(z, a) \rightarrow R_k(y, z) \]
\[ C_1(x) \land R_1(x, y) \land \cdots C_n(z) \land R_l(z, a) \rightarrow C_h(y) \]

\( C_i \) and \( R_i \) are concept and role names of the ontological KB

Examples:

- \( \text{Person}(x) \land \text{hasWellPayedJob}(x, y) \Rightarrow \text{Manager}(x) \)
- \( \text{Employee}(x) \land \text{worksAt}(x, z) \land \text{workForProject}(x, y) \land \text{projectSupervisor}(y, x) \Rightarrow \text{isCompanyManagerOf}(z, x) \)
Language Bias (ensuring decidability)

- **safety condition**: all variables in the head must appear in the body
- **connection**: atoms share at least one variable or constant
- interpretation under **$DL - Safety$** condition: all variables in the rule bind only to known individuals in the ontology
- **Non Redundancy**: there are no atoms that can be derived by other atoms

Example (Redundant Rule)

Given $K$ made by the TBox $T = \{Father \sqsubseteq Parent\}$ and the rule

$$r := Father(x) \land Parent(x) \rightarrow Human(x)$$

$r$ is redundant since $Parent(x)$ is entailed by $Father(x)$ w.r.t. $K$. 

Inspired to the general framework for discovering frequent *Datalog* patterns \[Dehaspe \textit{et al.}'99; Goethals \textit{et al.}'02\]

Grounded on a level-wise *generate-and-test* approach

- **Start**: initial general pattern i.e. a concept name (jointly with a variable name) or a role name (jointly with variable names)
- **Proceed**: at each level with
  - specializing the pattern by the use of suitable operators
  - evaluate the generated specializations for possible pruning
- **Stop**: stopping criterion met

A rule is a list of atoms (interpreted as a conjunction) where the first one represents the head \[Galarraga \textit{et al.} @WWW'13\]

The specialization operators represent the way for exploring the search space.
Pattern Specializations

For a given pattern all possible specializations are generated by applying the operators:

- **Add a concept atom**: adds an atom with a concept name as a predicate symbol and an *already appearing* variable as argument.
- **Add a role atom**: adds an atom with a role name as a predicate symbol; *at least one variable already appears* in the pattern.

The Operators are applied so that always *connected and non-redundant rules* are obtained.

Additional operators for taking into account constants could be similarly considered.
Pattern Specializations: Examples

Pattern to be Specialized \( C(x) \land R(x, y) \)

**Non Redundant Concept \( D \)**

Refined Patterns

1. \( C(x) \land R(x, y) \land D(x) \)
2. \( C(x) \land R(x, y) \land D(y) \)

**Non Redundant Role \( S \)**

Fresh Variable \( z \)

Refined Patterns

1. \( C(x) \land R(x, y) \land S(x, z) \)
2. \( C(x) \land R(x, y) \land S(z, x) \)
3. \( C(x) \land R(x, y) \land S(y, z) \)
4. \( C(x) \land R(x, y) \land S(z, y) \)

**Non Redundant Role \( S \)**

All Variables Bound

Refined Patterns

1. \( C(x) \land R(x, y) \land S(x, x) \)
2. \( C(x) \land R(x, y) \land S(x, y) \)
3. \( C(x) \land R(x, y) \land S(y, x) \)
4. \( C(x) \land R(x, y) \land S(y, y) \)
Exploitation of the Relational Association Rules and Utility

- **ABox completion**
  - rules may fire new assertions

- **Ontology Enrichment**
  - A rule may suggest an **inclusion axiom** that is missing in the ontology
    - *e.g.* \( \text{Car}(x) \Rightarrow \text{Vehicle}(x) \)
  - A rule may suggest a **disjointness axiom** that is missing in the ontology
    - *e.g.* \( \text{Man}(x) \Rightarrow \neg \text{Woman}(x) \)
  - A rule may suggest **symmetry** for a role that is missing in the ontology
    - *e.g.* \( \text{isFriendOf}(x, y) \Rightarrow \text{isFriendOf}(y, x) \)
  - A rule may suggest **transitivity** for a role that is missing in the ontology
    - *e.g.* \( \text{isTopicRelatedTo}(x, y) \land \text{isTopicRelatedTo}(y, z) \Rightarrow \text{isTopicRelatedTo}(x, z) \)

- **Creating Ontology with Enriched expressive power**
  - discovered rules can be straightforwardly integrated with the existing ontology
On Evaluating the Pattern Discovery Method

GOALS:

1. assessing the *ability of the discovered rules to predict* new assertional knowledge

2. showing the *value added of exploiting* background knowledge and reasoning capabilities when extracting rules

Publicly available ontologies used
### GOAL 1: Results/Lesson Learnt

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Sample Rate</th>
<th>Match Rate</th>
<th>Comm. Rate</th>
<th>Ind. Rate</th>
<th>Precision</th>
<th>Tot. nr. Predictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Financial</td>
<td>20%</td>
<td>0.81</td>
<td>0</td>
<td>0.19</td>
<td>1.0</td>
<td>947</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>0.81</td>
<td>0</td>
<td>0.19</td>
<td>1.0</td>
<td>1890</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>0.82</td>
<td>0</td>
<td>0.18</td>
<td>1.0</td>
<td>2960</td>
</tr>
<tr>
<td>BioPAX</td>
<td>20%</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>669</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>1059</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>1618</td>
</tr>
<tr>
<td>NTMerged</td>
<td>20%</td>
<td>0.94</td>
<td>0</td>
<td>0.06</td>
<td>1.0</td>
<td>9085</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>0.9</td>
<td>0</td>
<td>0.1</td>
<td>1.0</td>
<td>9756</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>0.94</td>
<td>0</td>
<td>0.06</td>
<td>1.0</td>
<td>10418</td>
</tr>
</tbody>
</table>

Note: Precision (does not considered induced results)

- **high match rate values** ⇒ **rules are able to predict new assertional knowledge**
- **null commission rate** ⇒ **no contradicting knowledge predicted**
- **induction rate not null** ⇒ **the developed method is able to induce new knowledge not logically derivable**
GOAL 2: Results/Lesson Learnt I

- system compared with AMIE [Galarraga et al. @WWW’13]
  - no use of background knowledge and reasoning capabilities

- compared number of discovered rules

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Samp.</th>
<th># Rules</th>
<th>Top</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Ours</td>
<td>AMIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Financial</td>
<td>20%</td>
<td>177</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>181</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>180</td>
<td>2</td>
</tr>
<tr>
<td>BioPax</td>
<td>20%</td>
<td>298</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>283</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>272</td>
<td>0</td>
</tr>
<tr>
<td>NTMerged</td>
<td>20%</td>
<td>243</td>
<td>1129</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>225</td>
<td>1022</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>239</td>
<td>1063</td>
</tr>
</tbody>
</table>

- outperformed the number of rules for Financial and BioPax
  - our system output rules having both concept and role atoms as head
  - our system can prune redundant and inconsistent rules and rules
  - reason why AMIE registered a larger number of rules for NTNmerged.
Issues/Lessons Learnt

Develop a **scalable** algorithm

- Exploiting Evolutionary-based approaches for outperforming the exploration of the search space [*d’Amato et al. @ EKAW 2016*]

Other directions

- *additional heuristics for reducing* the exploration of the *search space* and/or possible optimizations
- (New) metrics for the evaluation of the *interestingness of the discovered rules* (potential inner and post pruning)
Conclusions

**Machine Learning methods**
- could be usefully exploited for ontology mining
- suitable in case of incoherent/noisy KBs
- can be seen as an additional layer on top of deductive reasoning for realizing *new/additional forms of approximated reasoning capabilities*

**Future directions:**
- Semi-Supervised Learning methods particularly appealing for LOD
- Special focus on scalability issues
- Frequent Graph Patterns mining methods for the SW needs to be investigated
That’s all!

Thank you

Nicola Fanizzi  Steffen Staab  Floriana Esposito

Andrea Tettamanzi  Fabien Gandon  Tran Duc Minh