On the Need of Semantics when Tackling Knowledge Graph Refinement under a Machine Learning Perspective

Claudia d’Amato

Computer Science Department
University of Bari "Aldo Moro", Bari, Italy

20th International Conference on Principles of Knowledge Representation and Reasoning (KR 2023)

September 6, 2023
Introduction & Motivation

Open KG
online with content freely accessible
- BabelNet
- DBpedia
- Freebase
- Wikidata
- YAGO
- ....

Enterprise KG
for commercial usage
- Google
- Amazon
- Facebook
- LinkedIn
- Microsoft
- ....

1 picture from https://www.csee.umbc.edu/courses/graduate/691/fall19/07/
Applications

- e-Commerce
- Semantic Search
- Fact Checking
- Personalization
- Recommendation
- Medical decision support system
- Question Answering
- Machine Translation
- ...

Research Fields

- Information Extraction
- Natural Language Processing
- Machine Learning (ML)
- Knowledge Representation
- Web
- Robotics
- ...

C. d’Amato (UniBa)

Semantics, ML & KGs
Knowledge Graph: Definition [Hogan et al., 2021]

A graph of data intended to convey knowledge of the real world

- conforming to a graph-based data model
- nodes represent entities of interest
- edges represent different relations between these entities
- data graph potentially enhanced with schema

KGs: Main Features

- ontologies employed to define and reason about the semantics of nodes and edges
- RDF, RDFS, OWL representation languages largely adopted
- grounded on the Open World Assumption (OWA)
- very large data collections
- suffer of incompleteness and noise
  - since often result from a complex building process
Machine Learning & Knowledge Graphs
ML and KGs

Two perspectives:

**KG as input to ML**

**Goal:** improving the performance in many learning tasks, e.g.
- Question Answering (QA)
- image classification
- instance disambiguation
- text summarization
- ...

**ML as input to KG**

**Goal:** improving the KG itself
- creating new facts
- creating generalizations
- prototyping
- improving the size, coverage, depth and accuracy of KGs → reducing their production costs
Reason Why Semantics is Needed

Numeric-based methods

- highly scalable
- consist of series of numbers without any obvious human interpretation
  - no interpretable models provided
  - impact on interpretability, explainability, trustworthiness of results
- no background knowledge and reasoning capabilities generally exploited
  - only factual information is considered

- knowledge within KG
  - only partially considered
  - and not always in a fully correct way (negatives)

---


3 Picture from https://github.com/topics/knowledge-graph-embeddings
KG as input to ML

(Infusing KG into Large Language Models)
Large Language Models (LLMs)

LLM: a type AI algorithm that
- uses deep learning techniques and
- massively large data sets
- to understand, summarize, generate and predict new content
  - specifically architected to help generate text-based content
- typically has one billion or more parameters
- is able to understand and generate accurate responses rapidly

Modern LLMs
- use transformer neural networks (transformers)

LLMs take a complex approach that involves multiple components
Main Usages of Large Language Models

Once an LLM is trained, a base exists on which performing

- Text generation
- Translation (LLMs trained on multiple languages)
- Content categorization, Content summary and Rewriting
- Conversational AI and chatbots (e.g. ChatGTP)
- Sentiment analysis
LLM: Challenges and Limitations

- **Development costs**
  - running LLMs require large quantities of expensive graphics processing unit hardware and massive data sets

- **Operational costs** → can be very high for the hosting organization

- **Bias (since trained on unlabeled data)**
  - no guarantee that known bias is removed

- **Explainability** → almost missing

- **Hallucination**
  - when providing an inaccurate response not based on trained data

- **Complexity**
  - can be particularly complex to troubleshoot

- **Glitch tokens**
  - maliciously designed prompts that cause an LLM to malfunction
Infusing KG into LLMs

Infusing factual triples of KG into a LLM [Moiseev et al., 2022]

Goal: assess whether LLMs can better internalize knowledge from structured data (KG) or from text on QA tasks

Result: models pre-trained on KG outperform the baseline pre-trained on text sentences containing the same knowledge

Open Challenges

What would be the impact of additionally exploiting KG semantics (e.g. concept hierarchy) and reasoning capabilities (generating also additional fact triples)?

- additionally exploiting KG semantics (e.g. concept hierarchy)
- and reasoning capabilities (generating also additional fact triples)?

⇒ experimental study required
- a new solution for incorporating semantics would be needed
ML as input to KG

(KG Refinement by KG Embedding Methods)

(KG Refinement by Symbol-based Learning Methods)
KG Refinement by KG Embedding Methods:
Injecting Semantics
Incompleteness and noise

Knowledge Graph Refinement

- Link Prediction: predicts missing links between entities
  - regarded as a learning to rank problem
- Triple Classification: assesses correctness of a statement wrt a KG
  - regarded as a binary classification problem

Very Large Data Collections

New scalable Machine Learning methods

- grounded on numeric-based approaches
- KG vector embedding models (KGE) largely investigated [Cai et al., 2018]

ML/KGE for KGs: Issues

- CWA (or LCWA) mostly adopted vs. OWA
- schema level information and reasoning capabilities almost disregarded
- no interpretable models ⇒ hard to motivate results
KG Embedding Models...

KGE models convert data graph into an optimal low-dimensional space [Cai et al., 2018]

Graph structural information and properties preserved as much as possible

---

4 Picture from https://laptrinhx.com/node2vec-graph-embedding-method-2620064815/
KGE methods differ in their main building blocks [Ji et al., 2020]:

- **Representation Space**
  - where representing relations and entities
  - point-wise, complex, discrete, Gaussian, manifold, etc.

- **Encoding Model**
  - for representing and learning relational interactions
  - linear, factorization, neural models, etc.

- **Scoring Function**
  - for measuring the plausibility of factual triples
  - based on distance, energy, semantic matching, other criteria

- **Auxiliary Information**
  - to be incorporated into the embedding methods
  - text, type, images
KG Embedding Models

Goal

Learning embeddings s.t.

- score of a valid (positive) triple is higher than
- the score of an invalid (negative) triple

Negative examples generated by random corruption of triples

- false negatives may be generated
- only triple directly observable are considered

---

5 Picture from "ECAI-20 Tutorial: Knowledge Graph Embeddings: From Theory to Practice"

C. d’Amato (UniBa)  Semantics, ML & KGs  KR 2023  18 / 52
**Idea:** Enhance KGE through Background Knowledge (BK) Injection

By two components:

**Reasoning:** used for generating negative triples

**Axioms:** domain, range, disjointWith, functionalProperty;

**BK Injection:** defines constraints on functions, corresponding to the considered axioms, guiding the way embedding are learned

**Axioms:** equivClass, equivProperty, inverseOf and subClassOf.
Other KG Embedding Methods Leveraging BK

- Jointly embedding KGs and logical rules [Guo et al., 2016]
  - triples represented as atomic formulae
  - rules represented as complex formulae modeled by t-norm fuzzy logics
- Adversarial training exploiting Datalog clauses encoding assumptions to regularize neural link predictors [Minervini et al., 2017a]

A specific form of BK required, not directly applicable to KGs
An approach to learn embeddings exploiting BK [d’Amato et al., 2021c,b]

**TRANSOWL**

TransE

[Bordes et al., 2013]

**TRANSROWL**

TransR

[Lin et al., 2015]

**TRANSROWL^R**

Could be applied to more complex KG embedding methods with additional formalization
TransOWL

- Derive further triples to be considered for training via schema axioms
  - equivClass, equivProperty, inverseOf and subClassOf
- More complex loss function
  - adding a number of terms consistently with the constraints

\[
L = \sum_{\langle h, r, t \rangle \in \Delta} [\gamma + f_r(h, t) - f_r(h', t')]_+ + \sum_{\langle t, q, h \rangle \in \Delta_{\text{equivClass}}} [\gamma + f_q(t, h) - f_q(t', h')]_+ \\
+ \sum_{\langle h, s, t \rangle \in \Delta_{\text{equivProperty}}} [\gamma + f_s(h, t) - f_s(h', t')]_+ + \sum_{\langle h', \text{typeOf}, l \rangle \in \Delta_{\text{equivClass}}} [\gamma + f_{\text{typeOf}}(h, l) - f_{\text{typeOf}}(h', l')]_+ \\
+ \sum_{\langle h', \text{subClassOf}, p \rangle \in \Delta_{\text{equivClass}}} [((\gamma - \beta) + f(h, p) - f(h', p'))_+] \\
\]

where \( q \equiv r^- \), \( s \equiv r \) (properties), \( l \equiv t \) and \( t \sqsubseteq p \) (classes) and \( f(h, p) = \|e_h - e_p\| \)
TransROWL

- TransOWL loss function adopted plus *weighting parameters*
  - equivClass, equivProperty, inverseOf and subClassOf
- TransR score function adopted

\[
L = \sum_{\langle h, r, t \rangle \in \Delta} [\gamma + f'_r(h, t) - f'_r(h', t')]_+ + \lambda_1 \sum_{\langle t, q, h \rangle \in \Delta', \langle t', q, h' \rangle \in \Delta', \text{inverseOf}} [\gamma + f'_q(t, h) - f'_q(t', h')]_+ \\
+ \lambda_2 \sum_{\langle h, s, t \rangle \in \Delta, \langle h', s, t' \rangle \in \Delta', \text{equivProperty}} [\gamma + f'_s(h, t) - f'_s(h', t')]_+ + \lambda_3 \sum_{\langle h, \text{typeOf}, l \rangle \in \Delta, \langle h', \text{typeOf}, l' \rangle \in \Delta', \text{equivClass}} [\gamma + f'_{\text{typeOf}}(h, l) - f'_{\text{typeOf}}(h', l')]_+ \\
+ \lambda_4 \sum_{\langle t, \text{subClassOf}, p \rangle \in \Delta, \langle t', \text{subClassOf}, p' \rangle \in \Delta', \text{subClass}} [\gamma - \beta + f'(t, p) - f'(t', p')]_+
\]

where

- \( q \equiv r^-, s \equiv r \) (properties), \( l \equiv t \) and \( t \sqsubseteq p \) (classes)
- the parameters \( \lambda_i, i \in \{1, \ldots, 4\} \), weigh the influence that each function term has during
  the learning phase
An Alternative Approach: TransROWL$^R$

Adopting an **axiom-based regularization** of the loss function as for TransE$^R$ [Minervini et al., 2017b]

- by adding specific constraints to the loss function **rather than**
- explicitly derive additional triples during training

**Loss function**

\[
L = \sum_{\langle h, r, t \rangle \in \Delta} \left[ \gamma + f'_r(h, t) - f'_r(h', t') \right] + \\
+ \lambda_1 \sum_{r \equiv q^- \in T_{\text{inverseOf}}} \| r + q \| + \lambda_2 \sum_{r \equiv q^- \in T_{\text{inverseOf}}} \| M_r - M_q \| \\
+ \lambda_3 \sum_{r \equiv p \in T_{\text{equivProp}}} \| r - p \| + \lambda_4 \sum_{r \equiv p \in T_{\text{equivProp}}} \| M_r - M_p \| \\
+ \lambda_5 \sum_{e' \equiv e'' \in T_{\text{equivClass}}} \| e' - e'' \| + \lambda_6 \sum_{s' \subseteq s'' \in T_{\text{subClass}}} \| 1 - \beta - (s' - s'') \|
\]
Lesson Learnt from Experiments...

**Goal:** Assessing the benefit of exploiting BK

- Comparing\(^6\) TransOWL, TransROWL, TransROWL\(^R\) over to the original models TransE and TransR as a baseline

**Performances tested on:**

- Link Prediction task
- Triple Classification task
- Standard metrics adopted

**KGs adopted:**

<table>
<thead>
<tr>
<th>KG</th>
<th>#Triples</th>
<th>#Entities</th>
<th>#Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBpedia15K</td>
<td>180000</td>
<td>12800</td>
<td>278</td>
</tr>
<tr>
<td>DBpedia100K</td>
<td>600000</td>
<td>100000</td>
<td>321</td>
</tr>
<tr>
<td>DBpediaYAGO</td>
<td>290000</td>
<td>88000</td>
<td>316</td>
</tr>
<tr>
<td>NELL(^7)</td>
<td>150000</td>
<td>68000</td>
<td>272</td>
</tr>
</tbody>
</table>

\(^6\) All methods implemented as publicly available systems [https://github.com/Keehl-Mihael/TransROWL-HRS](https://github.com/Keehl-Mihael/TransROWL-HRS)

\(^7\) equivalentClass and equivalentProperty missing; limited number of typeOf-triples; abundance of subClassOf-triples
...Lesson Learnt from Experiments

- Best performance achieved by TransROWL, in most of the cases, and TransROWL$^R$
- TransROWL slightly superior performance of TransROWL$^R$

- NELL was aimed at testing in condition of larger incompleteness
  - equivalentClass and equivalentProperty missing
  - low number of typeof-triples per entity
  - the models showed oscillating performances wrt the baselines

Open Challenges

- further enhance semantic KGE models with additional schema axioms
- extend the framework to more complex KGE models
- improve semantic KGE models for coping with incompleteness → further experiments needed
Exploiting Semantics for Providing
Explanations to Link Predictions on KGs
A-Posteriori Explanations of Link Predictions...

**A-posteriori methods** find suitable explanation(s) based on the observed output and the model input, independently on the KGE adopted.

Given the predicted triple: \( \langle \text{NickMason}, \text{recordLabel}, \text{CapitolRecords} \rangle \) why is it provided?

User is able to understand motivations, and trust (or not) the prediction.

**Example of explanation**

\[
\langle \text{NickMason}, \text{associatedBand}, \text{PinkFloyd} \rangle, \\
\langle \text{PinkFloyd}, \text{recordLabel}, \text{CapitolRecords} \rangle
\]

Ideally supported by analogous situations to be found in the KG e.g. \( \langle \text{RingoStarr}, \text{recordLabel}, \text{Parlophone} \rangle \) for which the computed explanation is:

\[
\langle \text{RingoStarr}, \text{associatedBand}, \text{TheBeatles} \rangle, \\
\langle \text{TheBeatles}, \text{recordLabel}, \text{Parlophone} \rangle.
\]

C. d’Amato (UniBa) 
Semantics, ML & KGs 
KR 2023
A-**posteriori methods**: developed solutions

**KELPIE** [Rossi *et al.*, 2022]: generates necessary and sufficient (path) conditions and an articulated new evaluation protocol

**CrossE** [Zhang *et al.*, 2019]: embedding model for link predictions providing explanations

- the search for a path linking the subject *h* and object *t* of a predicted triple \( \langle h, r, t \rangle \)
  - **Max length 2** → six types of paths possible:
    - Length 1: \( P_1 = \{ \langle h, r_s, t \rangle \} \), \( P_2 = \{ \langle t, r_s, h \rangle \} \)
    - Length 2: \( P_3 = \{ \langle e', r_s, h \rangle, \langle e', r', t \rangle \} \), \( P_4 = \{ \langle e', r_s, h \rangle, \langle t, r', e' \rangle \} \), \( P_5 = \{ \langle h, r_s, e' \rangle, \langle e', r', t \rangle \} \), \( P_6 = \{ \langle h, r_s, e' \rangle, \langle t, r', e' \rangle \} \),
    - where \( r_s \) similar to \( r, r' \) any other relationship, \( e' \) any other entity;

- search driven by similarities between relation/entity embeddings by Euclidean distance

- **structural comparisons** with other paths in the KG to reinforce the reliability of the explanation found (referred to as support)
**Idea:** **SemanticCrossE** provide semantic-based explanations for link predictions on KGs

Extends CrossE by adopting a Semantic Cosine similarity that leads the explanation process

- exploits the underlying KG semantics → Domain, Range and Classes considered
- increases the cosine similarity of two (entities / relationships) vector embeddings on the ground of available additional semantic information which is captured by a semantic Score function

**Definition (semantic Cosine)**

Given KG $\mathcal{K}(\mathcal{E}, \mathbb{R})$, the semantic Cosine measure for two entities $e, e' \in \mathcal{E}$ is defined by:

$$\text{semCos}_{\alpha, \beta}(e, e') = \alpha \cdot \text{sScore}(e, e') + \beta \cdot \text{sim}_\cos(e, e')$$

where $e$ the respective entity embedding vector; $\alpha, \beta \in [0, 1]$ chosen s.t. $\alpha + \beta = 1$.

In the case of relations $r, r' \in \mathbb{R}$ the measure is defined analogously.
Definition (semantic Score)

Given $C$ set of the classes occurring in $\mathcal{K}(\mathcal{E}, \mathbb{R})$, and the functions $Cl: \mathcal{E} \rightarrow C$, $Do: \mathbb{R} \rightarrow C$, and $Ra: \mathbb{R} \rightarrow C$ that return, resp., the conjunction of the classes an entity belongs to, and the domain and range of a relation, the semantic Score function for pairs of entities $e, e' \in \mathcal{E}$ is defined by:

$$sScore(e, e') = \frac{|\text{ret}[Cl(e) \cap Cl(e')]|}{|\text{ret}[Cl(e) \cup Cl(e')]|}$$

where $\text{ret}_\mathcal{K}(C)$ returns the entities that can be proven to belong to a given class $C$

Analogously, given any two relationships $r, r' \in \mathbb{R}$, it is defined:

$$sScore(r, r') = \frac{|\text{ret}[Do(r) \cap Do(r')]|}{|\text{ret}[Do(r) \cup Do(r')]|} + \frac{|\text{ret}[Ra(r) \cap Ra(r')]|}{|\text{ret}[Ra(r) \cup Ra(r')]|}$$

- Approximated form of semantic Cosine measure (specifically of the semantic Score function) employed [d’Amato et al., 2021a]
- Efficient computation obtained by a preliminary clustering phase [d’Amato et al., 2023]
Lesson Learnt from Experiments...

**Goal:** Establishing the *impact of an added semantic component when computing explanations* of link prediction results

- Comparing ApproxSemanticCrossE and cosineCrossE to CrossE as baseline
- Code and datasets publicly available\(^8\)

<table>
<thead>
<tr>
<th>KG</th>
<th>#Triples</th>
<th>#Entities</th>
<th>#Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>FB15k-237</td>
<td>310116</td>
<td>14541</td>
<td>237</td>
</tr>
<tr>
<td>WN18</td>
<td>151442</td>
<td>40943</td>
<td>18</td>
</tr>
<tr>
<td>DBpedia15K</td>
<td>183218</td>
<td>12862</td>
<td>279</td>
</tr>
</tbody>
</table>

- KGs adopted by CrossE considered
- *DBpedia15K* additionally taken for testing the semantic component
- CrossE adopted for the preliminary link prediction phase
- Number \(k\) of similar relations; \(j\) of most similar entities; \(k = j = 3\)
- Weights for the approximated semantic cosine similarity: \(\alpha = 0.2; \beta = 0.8\)

\(^8\) [https://github.com/pierulohacker/SemanticCrossE/tree/master/explanation](https://github.com/pierulohacker/SemanticCrossE/tree/master/explanation)
...Lesson Learnt from Experiments

Metrics (as for CrossE):

- **Recall**: ratio of triples for which the model can generate explanations
- **Average Support**: number of explanations, on average, for each prediction
- **Recall** and **average support** computed for each of the 6 types of explanation path
- Different ratios (2% and 5%) of predictions considered for building explanations, starting from those ranking higher in the link prediction results

Results:

- ApproxSemanticCrossE showed improved results both in terms of recall and support
- ApproxSemanticCrossE not affected by noisy (irrelevant) explanations as for CrossE and partially cosineCrossE → qualitative evaluation conducted

Open Challenges

- Taking into account additional semantics in KGs (e.g. transitivity, symmetry etc.)
- Injecting BK within KELPIE framework
- Develop a standardized evaluation protocol for explanations
KG Refinement by Symbol-based Methods:

Learning Disjointness Axioms
Symbol-based methods

- usually provide interpretable models
  - e.g. trees, rules, logical formulae, etc.
  - may be exploited for a better understanding of the results
- could be able to exploit a background knowledge
- may be combined with deductive reasoning (e.g. to make predictions)
- limited in scalability

9 Picture from https://jaipancholi.com/model-interpretablity
Missing Disjointness Axioms: Issues

Disjointness axioms often missing [Wang et al., 2006]

Problems:

- introduction of noise

\[ \mathcal{K} = \{ \text{JournalPaper} \sqsubseteq \text{Paper}, \text{ConferencePaper} \sqsubseteq \text{Paper}, \text{ConferencePaper}(a), \text{Author}(a) \} \]

\( \mathcal{K} \) is consistent !!!

**Cause Axiom:** \( \text{Author} \equiv \neg \text{ConferencePaper} \text{ missing} \)

- counterintuitive inferences

\[ \mathcal{K} = \{ \text{JournalPaper} \sqsubseteq \text{Paper}, \text{ConferencePaper} \sqsubseteq \text{Paper}, \text{ConferencePaper}(a) \} \]

\( \mathcal{K} \models \text{JournalPaper}(a) ? \)

**Answer:** Unknown

**Cause Axiom:** \( \text{JournalPaper} \equiv \neg \text{ConferencePaper} \text{ missing} \)

- hard collecting negative examples when adopting numeric approaches
**Observation:** extensions of disjoint concepts do not overlap

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

**Idea:** Exploiting (Conceptual) clustering methods for mining disjointness axioms

---

**Definition (Problem Definition)**

**Given**

- an ontological knowledge base $\mathcal{K} = \langle T, A \rangle$
- a set of individuals (aka entities) $I \subseteq \text{Ind}(A)$

**Find**

- $n$ pairwise disjoint clusters $\{ C_1, \ldots, C_n \}$
- for each $i = 1, \ldots, n$, a concept description $D_i$ that describes $C_i$, such that:
  - $\forall a \in C_i : \mathcal{K} \models D_i(a)$
  - $\forall b \in C_j, j \neq i : \mathcal{K} \models \neg D_i(b)$.
  - Hence $\forall D_i, D_j, i \neq j : \mathcal{K} \models D_j \sqsubseteq \neg D_i$.  

[Rizzo et al., 2021]
Learning Disjointness Axioms: Developed Methods

Statistical-based approach

- NAR - exploiting negative association rules [Fleischhacker and Völker, 2011]
- PCC - exploiting Pearson’s correlation coeff. [Völker et al., 2015]

Do not exploit any background knowledge and reasoning capabilities

Disjointness axioms learning/discovery can be hardly performed without symbol-based methods
Terminological Cluster Tree

Defined a method\(^{10}\) for eliciting disjointness axioms [Rizzo et al., 2021]

- solving a clustering problem via learning Terminological Cluster Trees
- providing a concept description for each cluster

**Definition (Terminological cluster tree (TCT))**

A binary logical tree where

- a leaf node stands for a cluster of individuals \(C\)
- each inner node contains a description \(D\) (over the signature of \(\mathcal{K}\))
- each departing edge corresponds to positive (left) and negative (right) examples of \(D\)

---

\(^{10}\) Implemented system publicly available at [https://github.com/Giuseppe-Rizzo/TCTnew](https://github.com/Giuseppe-Rizzo/TCTnew)
Example of TCT

Given $I \subseteq \text{Ind}(A)$, an example of TCT describing the AI research community.
Collecting Disjointness Axioms

Given a TCT \( T \):

Step I:
- Traverse the \( T \) to collect the concept descriptions describing the clusters at the leaves
- A set of concepts \( CS \) is obtained

Step II:
- A set of candidate axioms \( A \) is generated from \( CS \):
  - an axiom \( D \sqsubseteq \neg E \) (\( D, E \in CS \)) is generated if
    - \( D \not\equiv E \) (or \( D \not\sqsubseteq E \) or viceversa - reasoner needed)
    - \( E \sqsubseteq \neg D \) has not been generated
Inducing a TCT

Given the set of individuals \( I \) and \( \top \) concept

Divide-and-conquer approach adopted

- **Base Case:** test the stopCondition
  - the cohesion of the cluster \( I \) exceeds a threshold \( \nu \)
  - distance between medoids below a threshold \( \nu \)

- **Recursive Step** (stopCondition does not hold):
  - a set \( S \) of refinements of the current (parent) description \( C \) generated
  - the bestConcept \( E^* \in S \) is selected and installed as current node
    - the one showing the best cluster separation \( \Leftrightarrow \) with max distance between the medoids of its positive \( P \) and negative \( N \) individuals

- \( I \) is split in:
  - \( I_{\text{left}} \subseteq I \Leftrightarrow \) individuals with the smallest distance wrt the medoid of \( P \)
  - \( I_{\text{right}} \subseteq I \Leftrightarrow \) individuals with the smallest distance wrt the medoid of \( N \)
  - reasoner employed for collecting \( P \) and \( N \)

**Note:** Number of clusters not required - obtained from data distribution
Lesson Learnt from experiments

Experiments performed on ontologies publicly available

- **Goal I**: Re-discover a target axiom (existing in $\mathcal{K}$)
  - **Metrics**: # discovered axioms and #cases of inconsistency
  - **Results**:
    - target axioms rediscovered for almost all cases
    - additional disjointness axioms discovered in a significant number
    - limited number of inconsistencies found

<table>
<thead>
<tr>
<th>Ontology</th>
<th>DL Language</th>
<th>#Concepts</th>
<th>#Roles</th>
<th>#Individuals</th>
<th>#Disj. Ax.s</th>
</tr>
</thead>
<tbody>
<tr>
<td>BioPax</td>
<td>$\mathcal{ALCIF}(D)$</td>
<td>74</td>
<td>70</td>
<td>323</td>
<td>85</td>
</tr>
<tr>
<td>NTN</td>
<td>$\mathcal{SHIF}(D)$</td>
<td>47</td>
<td>27</td>
<td>676</td>
<td>40</td>
</tr>
<tr>
<td>Financial</td>
<td>$\mathcal{ALCIF}(D)$</td>
<td>60</td>
<td>16</td>
<td>1000</td>
<td>113</td>
</tr>
<tr>
<td>GeoSkills</td>
<td>$\mathcal{ALCHOIN}(D)$</td>
<td>596</td>
<td>23</td>
<td>2567</td>
<td>378</td>
</tr>
<tr>
<td>Monetary</td>
<td>$\mathcal{ALCHIF}(D)$</td>
<td>323</td>
<td>247</td>
<td>2466</td>
<td>236</td>
</tr>
<tr>
<td>DBPedia3.9</td>
<td>$\mathcal{ALCHI}(D)$</td>
<td>251</td>
<td>132</td>
<td>16606</td>
<td>11</td>
</tr>
</tbody>
</table>
Lesson Learnt from experiments [...cont.]

Goal II:
- Re-discover randomly selected target axioms added according to the **Strong Disjointness Assumption** [Schlobach, 2005]
  - two sibling concepts in a subsumption hierarchy considered as disjoint
- **comparative** analysis with **statistical-based** methods:
  - PCC - based on Pearson’s correlation coefficient [Völker et al., 2015]
  - NAR - exploiting **negative association** rules [Fleischhacker and Völker, 2011]
- Setting:
  - A copy of each ontology created removing 20%, 50%, 70% of the disjointness axioms
- **Metrics**: rate of **rediscovered** target axioms, #cases of inconsistency, # additional discovered axioms
Lesson Learnt from experiments [...cont.]

- Results:
  - almost all axioms rediscovered
    - Rate decreases when larger fractions of axioms removed, as expected
  - TCT outperforms PCC and NAR wrt additionally discovered axioms whilst introducing limited inconsistency
    - TCT allows to express complex disjointness axioms
    - PCC and NAR tackle only disjointness between concept names

Exploiting $\mathcal{K}$ as well as the data distribution improves disjointness axioms discovery

Open Challenge

Develop an extensive experimental user study on the validity and significance of the complex disjointness axioms discovered by TCT
Conclusions
Conclusions:

- **Injecting semantics** and exploiting reasoning capabilities **may improve the effectiveness of ML solutions for KG**
  - Framework for injecting BK into KGE models
  - Solution for injecting semantics when computing a-posteriori explanations to link predictions
- **Symbol-based methods useful for supplementing schema level information**
  - Conceptual Clustering for Learning Disjointness Axioms

Next Research Challenges:

- Extend the framework for injecting BK to more complex KGE models
- Empower the semantic explanation process with additional schema axioms
- Scalability of symbol-based learning methods still need to be improved
- New solutions required for enhancing LLMs with KG semantics
Thank you

Nicola Fanizzi
Giuseppe Rizzo
Giovanni Sansaro
Nicola Flavio Quatraro
Pierpaolo Masella
Francesco Benedetti
References


References

...cont.


References […cont.]


