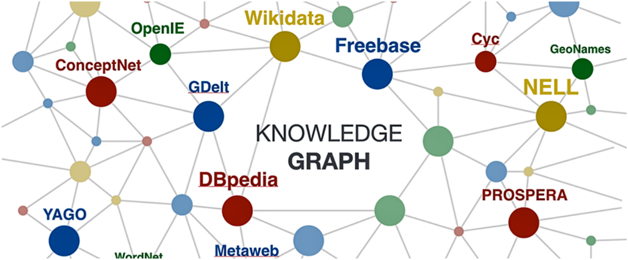
Empowering Knowledge Bases: A Machine Learning Perspective

Claudia d'Amato

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

34th International Workshop on Description Logics

22 September 2021



[1]

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 Areas

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

Knowledge Graph: Definition

[a] 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 potentially different relations between these entities
- data graph potentially enhanced with schema

[a] A. Hogan et al. Knowledge Graphs. ACM Comput. Surv. 54(4): 71:1-71:37 (2021)

KGs: Main Features

- *ontologies* employed to define and reason about the semantics of nodes and edges
- RDF, RDFS, OWL representation languages will be assumed
- grounded on the Open World Assumption (OWA)
- very large data collections

Issues

- KG suffer of *incompleteness* and *noise*
  - e.g. missing links, wrong links
  - since often result from a complex building process
- Ontologies and assertions can be out-of-sync
  - resulting incomplete, noisy and sometimes inconsistent wrt the actual usage of the conceptual vocabulary in the assertions
- Reasoning cannot be performed or may return counterintuitive results

Machine Learning methods adopted to discover new/additional knowledge by exploiting *the evidence coming from the data [d'Amato et al. 2010, d'Amato 2020 @ SWJ]*

Symbol-based methods

- able to exploit background knowledge and (deductive) reasoning capabilities

$\Downarrow$

Ontology Mining

- *All activities that allow for discovering hidden knowledge from ontological KBs*

Numeric-based methods

- highly scalable

$\Downarrow$

Knowledge Graph Refinement

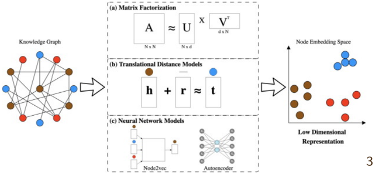
- *Link Prediction*: predicts missing links between entities
- *Triple Classification*: assesses statement correctness in a KG

[d'Amato 2020 @ SWJ] [2]

[2] C. d'Amato: Machine Learning for the Semantic Web: Lessons learnt and next research directions. Semantic Web 11(1): 195-203 (2020)

Numeric-based methods

- scalable
- consist of series of numbers without any obvious human interpretation

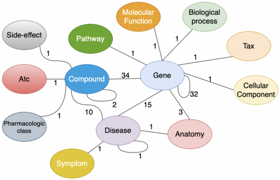


3

This may affects:

- the *interpretability* of the results
- the *explainability*
- and thus also somehow the *trustworthiness* of results

DRKG – Drug Repurposing Knowledge Graph



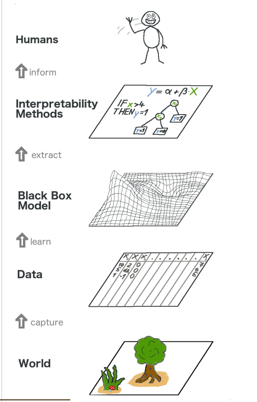
4

[3] Picture from D. N. Nicholson et al. Constructing knowledge graphs and their biomedical applications, Computational and Structural Biotechnology Journal, Vol. 18, pp. 1414–1428, (2020) ISSN 2001-0370

[4] Picture from https://github.com/topics/knowledge-graph-embeddings

Symbol-based learning methods usually provide

- *interpretable models* generalizing conclusions
  - e.g. trees, rules, logical formulae, etc.
- may be exploited for a better understanding of the provided results
- could be combined with deductive reasoning capabilities
- more limited ability to scale



5

[5] Picture from https://jaipancholi.com/model-interpretability

Symbol-based learning methods for:

- Concept Learning
- Learning Disjointness Axioms

Symbol-based learning methods for:

- Concept Learning
- Learning Disjointness Axioms

Semantic and validating schemata require domain experts for definitions and constraints.

Latent patterns in the data graph could be exploited

Goal: **a)** Learning descriptions for a given concept name / expression

Example : $\text{Man} \equiv \text{Human} \sqcap \text{Male}$

b) Learning descriptions for characterizing a given set of individuals

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

Idea: Regard the problem as a *supervised concept learning* task

Supervised Concept Learning:

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

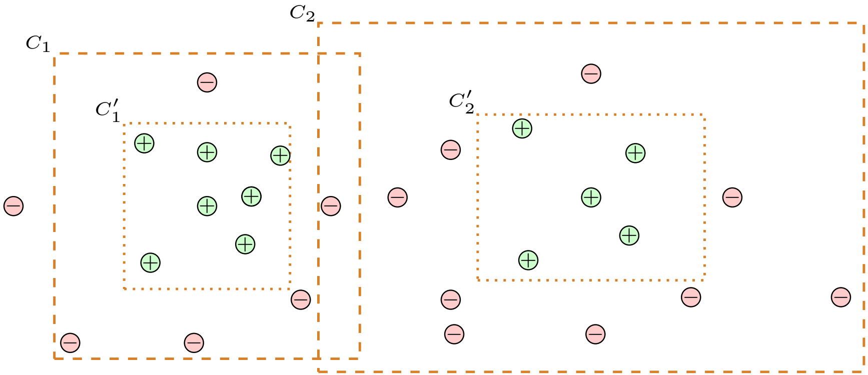
Definition (Problem Definition)

- *Given*
  - a knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$
  - a subset *pos* of individuals as positive examples of C
  - a subset *neg* of individuals as negative examples of C
- *Learn*
  - a DL concept description D so that
  - the individuals in *pos* are instances of D while those in *neg* are not

Developed Methods for Supervised Concept Learning

- **Separate-and-conquer approach**
  - YinYang *[Iannone et al. @ Appl. Intell. J. 2007]*
  - DL-FOIL *[Fanizzi et al. @ ILP 2008, Rizzo et al. @ FGCSJ 2020]*
  - DL-Learner *[Lehmann et al. @ MLJ 2010, SWJ 2011]*
- **Divide-and-conquer approach**
  - TermiTIS *[Fanizzi et al. @ ECML 2010, Rizzo et al. @ ESWC 2015, Rizzo et al. @ Aprox. Reas. J. 2018]*

DL-FOIL - Separate and Conquer: Example



C_1 = MasterStudent $\quad C'_1$ = MasterStudent $\sqcap \exists$worskIn.$\top$

C_2 = BachelorStudent $\quad C'_2$ = BachelorStudent $\sqcap \exists$worskIn.$\top$

Examples of Concept Descriptions Learnt with DL-FOIL

BioPax

induced:

```
Or( And( physicalEntity protein) dataSource)
```

original:

```
Or( And( And( dataSource externalReferenceUtilityClass)
ForAll(ORGANISM ForAll(CONTROLLED phys icalInteraction)))
protein)
```

NTN

induced:

```
Or( EvilSupernaturalBeing Not(God))
```

original:

```
Not(God)
```

Financial

induced:

```
Or( Not(Finished) NotPaidFinishedLoan Weekly)
```

original:

```
Or( LoanPayment Not(NoProblemsFinishedLoan))
```

Lesson Learnt from Experiments

- Relatively small ontological KBs adopted $\Rightarrow$ *scalability needs to be improved*
- Suitable concept descriptions learned $\Rightarrow$ *validation by expert recommended for adding axioms to the KB*
  - approximated descriptions may be learned depending of the threshold

Symbol-based learning methods for:

- Concept Learning
- Learning Disjointness Axioms

A fine grained schema level information can bring better insight of the data

Disjointness axioms often missing [6]

Problems:

- introduction of noise

$\mathcal{K} = \{ JournalPaper \sqsubseteq Paper,\ ConferencePaper \sqsubseteq Paper,\ ConferencePaper(a), Author(a)\ \}$
$\mathcal{K}$ is Consistent !!!
Cause Axiom: $Author \equiv \neg ConferencePaper$ **missing**

- counterintuitive inferences

$\mathcal{K} = \{ JournalPaper \sqsubseteq Paper,\ ConferencePaper \sqsubseteq Paper,\ ConferencePaper(a)\ \}$

$\mathcal{K} \models JournalPaper(a)?$
Answer: Unknown
Cause Axiom: $JournalPaper \equiv \neg ConferencePaper$ **missing**

- hard collecting negative examples when adopting numeric approaches

[6] Wang, T.D., Parsia, B., Hendler, J.: A survey of the web ontology landscape. In: Cruz, I., et al. (eds.) The Semantic Web - ISWC 2006, 5th Int. Semantic Web Conference Proceedings. LNCS, vol. 4273. Springer (2006), doi: 10.1007/11926078 49

Observation: extensions of disjoint concepts do not overlap

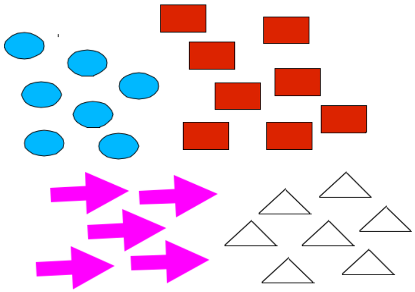
Question: would it be possible to *automatically capture* disjointness axioms by analyzing the data configuration/distribution?

Idea: Exploiting **(Conceptual) clustering methods** for the purpose

Clustering Methods

Unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

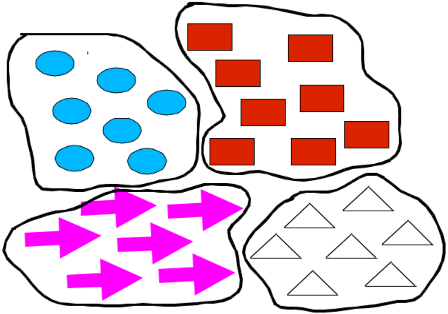
- intra-cluster *similarity* is high
- inter-cluster *similarity* is low



Clustering Methods

Unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

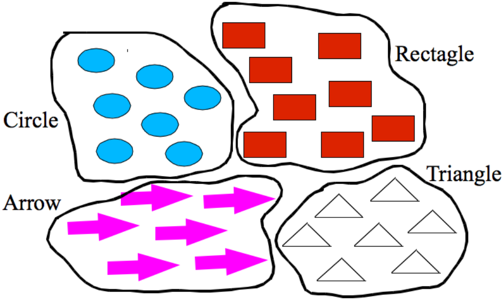
- intra-cluster *similarity* is high
- inter-cluster *similarity* is low



Clustering 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



Observation: extensions of disjoint concepts do not overlap

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

Idea: Exploiting **(Conceptual) clustering methods** for the purpose

Definition (Problem Definition)

Given

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

Find

- n pairwise disjoint clusters $\{\mathsf{C}_1, \ldots, \mathsf{C}_n\}$
- for each $i = 1, \ldots, n$, a concept description D_i that describes C_i, such that:
  - $\forall a \in \mathsf{C}_i : \mathcal{K} \models D_i(a)$
  - $\forall b \in \mathsf{C}_j, j \neq i : \mathcal{K} \models \neg D_i(b)$.
- Hence $\forall D_i, D_j, i \neq j : \mathcal{K} \models D_j \sqsubseteq \neg D_i$.

Learning Disjointness Axioms: Developed Methods

Statistical-based approach

- NAR - exploiting negative association rules *[Fleischhacker et al. @ OTM'11]* [7]
- PCC - exploiting Pearson's correlation coeff. *[Völker at al.@JWS 2015]* [8]

do not exploit any background knowledge and reasoning capabilities

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

[7] Fleischhacker, D. and Voelker, J. (2011). Inductive learning of disjointness axioms. In On the Move to Meaningful Internet Systems: OTM 2011 - Confederated International Conferences: CoopIS, DOA-SVI, and ODBASE 2011, Proceedings, Part II, volume 7045 of LNCS, pp. 680–697, Springer

[8] Voelker, J., Fleischhacker, D., and Stuckenschmidt, H. (2015). Automatic acquisition of class disjointness. J. of Web Semantics, 35, pp. 124–139

Terminological Cluster Tree

Defined a method [9] for eliciting disjointness axioms *[Rizzo et.al.@ SWJ'21]* [10]

- 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

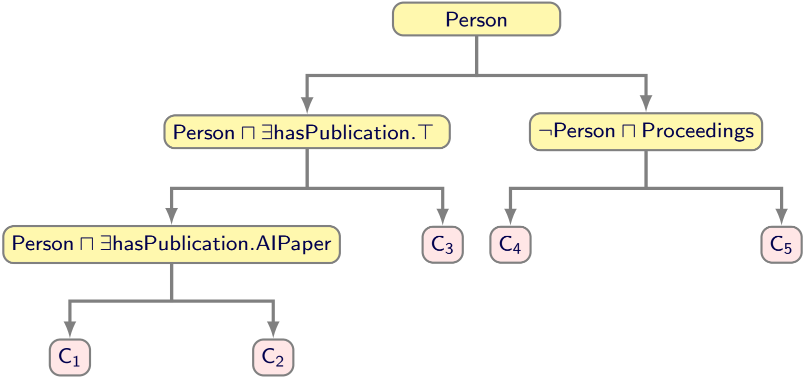
- a leaf node stands for a cluster of individuals C
- each inner node contains a description D (over the signature of $\mathcal{K}$)
- each departing edge corresponds to positive (left) and negative (right) examples of D

[9] Implemented system publicly available at `https://github.com/Giuseppe-Rizzo/TCTnew`

[10] G. Rizzo, C. d'Amato, N. Fanizzi: An unsupervised approach to disjointness learning based on terminological cluster trees. Semantic Web 12(3): 423-447 (2021)

Example of TCT

Given $\mathsf{I} \subseteq \mathsf{Ind}(\mathcal{A})$, an example of TCT describing the AI research community



Collecting Disjointness Axioms

Given a TCT T:

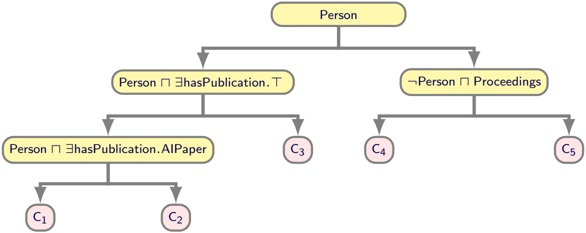
Step I:

- 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

Collecting Disjointness Axioms: Example



$$
\begin{aligned}
CS = \{ \quad & \text{Person}, \\
& \text{Person} \sqcap \exists \text{hasPublication}.\top, \\
& \neg(\text{Person} \sqcap \exists \text{hasPublication}.\top) \\
& \text{Person} \sqcap \exists \text{hasPublication}.\text{AIPaper} \\
& \neg\text{Person} \sqcap \text{Proceedings} \cdots \}
\end{aligned}
$$

Axiom1: $Person \sqcap \exists hasPublication.AIPaper \sqsubseteq \neg(\neg Person \sqcap Proceedings)$

Axiom2: $\cdots$

Inducing a TCT

Given the set of individuals I and $\top$ concept

Divide-and-conquere approach adopted

- **Base Case:** test the STOPCONDITION
  - the cohesion of the cluster I exceeds a threshold ν
    - distance between *medoids* below a threshold ν
- **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:
    - $I_{left} \subseteq$ I $\leftrightarrow$ individuals with the smallest distance wrt the *medoid* of P
    - $I_{right} \subseteq$ 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 I

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

Lesson Learnt from experiments II

Goal II:

- Re-discover randomly selected target axioms added according to the **Strong Disjointness Assumption** *[Schlobach et al. @ ESWC 2005]* [11]
    - two sibling concepts in a subsumption hierarchy considered as disjoint
- comparative analysis with statistical-based methods: PCC *[Völker at al. @ JWS 2015*, NAR *Fleischhacker et al. @ OTM'11]*
- Setting:
    - A copy of each ontology created removing 20%, 50%, 70% of the disjointness axioms
    - **Metrics**: rate of **rediscovered** target axioms, #cases of inconsistency, # addional discovered axioms

Lesson Learnt from experiments III

- 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**

[11] Schlobach, S. (2005). Debugging and semantic clarification by pinpointing. In The Semantic Web: Research and Applications, ESWC 2005, Proceedings, Vol. 3532, LNCS, pp. 226–240, Springer

Example of axioms

Successfully discovered axioms

- $\text{ExternalReferenceUtilityClass} \sqcap \exists \text{TAXONREF}.\top$
  disjoint with
  xref
- Activity
  disjoint with
  $\text{Person} \sqcap \exists \text{nationality}.\text{United\_states}$
- $\text{Person} \sqcap \text{hasSex}.\text{Male}$ ($\equiv$ Man)
  disjoint with
  $\text{SupernaturalBeing} \sqcap \text{God}$ ($\equiv$ God)

Not discovered axioms

- Actor disjoint with Artefact

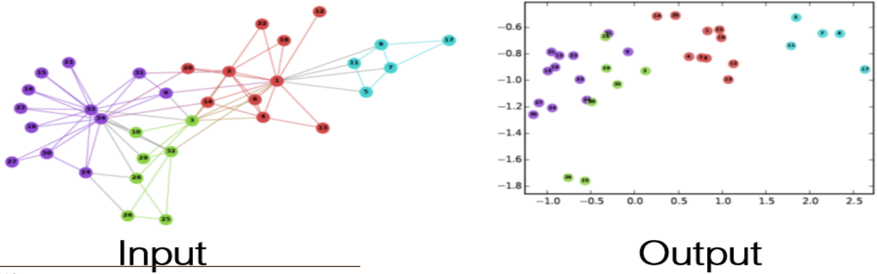
(concepts with few instances)

Numeric-based Methods for Knowledge Graph Refinement

KG Embedding Models...

Vector embedding models largely investigated [12]

- convert data graph into an optimal low-dimensional space
- *Graph structural information* preserved as much as possible
- CWA (or LCWA) mostly adopted vs. OWA
- *schema level information* and *reasoning* capabilities almost disregarded



[13]

[12] Cai, H. et al.: A comprehensive **survey** of graph embedding: problems, techniques, and applications. IEEE TKDE 30(09), pp. 1616-1637 (2018).

[13] Picture from https://laptrinhx.com/node2vec-graph-embedding-method-2620064815/

...KG Embedding Models...

Graph embedding methods differ in their main building blocks: [14]

the representation space: point-wise, complex, discrete, Gaussian, manifold, etc.

the encoding model: linear, factorization, neural models, etc.

the scoring function: based on distance, energy, semantic matching, other criteria, etc.

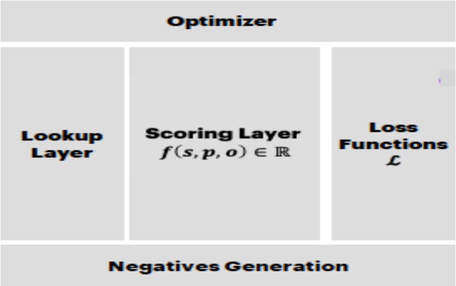
[14] Ji, S., Pan, S., Cambria, E., Marttinen, P., and Yu, P. (2021). A survey on knowledge graphs: representation, acquisition, and applications. IEEE Transactions on Neural Networks and Learning Systems.

...KG Embedding Models

Goal

Learning embeddings s.t.

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



[15]

[15] Picture from "ECAI-20 Tutorial: Knowledge Graph Embeddings: From Theory to Practice"

Idea: Enhance KGE through Background Knowledge Injection

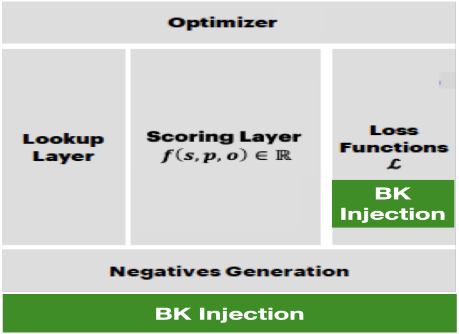
By two components:

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, S. et al. @ ACL 2016]* [16]
  - 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, P. et al. @ UAI 2017]* [17]

A specific form of BK required, not directly applicable to KGs

[16] Guo, S., Wang, Q., Wang, L., Wang, B., and Guo, L. (2016). Jointly embedding knowledge graphs and logical rules. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 192–202, Association for Computational Linguistics.

[17] Minervini, P., Demeester, T., Rocktaeschel, T., and Riedel, S. (2017). Adversarial sets for regularising neural link predictors. In UAI 2017 Proceedings. AUAI Press.

An approach to learn embeddings exploiting BK

[d'Amato et al. @ ESWC 2021] [18]

TransOWL

TransE

TransROWL TransROWL^R

TransR

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

[18] C. d'Amato, N. F. Quatraro, N. Fanizzi: Injecting Background Knowledge into Embedding Models for Predictive Tasks on Knowledge Graphs. ESWC 2021: 441-457 (2021)

TransOWL...

TransOWL maintains TransE setting

TransE[19] learns the vector embedding by minimizing

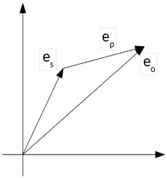
Margin-based loss function

$$L = \sum_{\substack{\langle s,p,o\rangle \in \Delta \\ \langle s',p,o'\rangle \in \Delta'}} [\gamma + f_p(\mathrm{e}_s, \mathrm{e}_o) - f_p(\mathrm{e}_{s'}, \mathrm{e}_{o'})]_+$$

where $[x]_+ = \max\{0, x\}$, and $\gamma \geq 0$

Score function

similarity (negative L_1 or L_2 distance) of the translated subject embedding $(\mathrm{e}_s + \mathrm{e}_p)$ to the object embedding e_o:



$$f_p(\mathrm{e}_s, \mathrm{e}_o) = -\|(\mathrm{e}_s + \mathrm{e}_p) - \mathrm{e}_o\|_{\{1,2\}}.$$

[19] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. Proceedings of NIPS 2013 (2013)

...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

$$
\begin{aligned}
L = {} & \overbrace{\sum_{\substack{\langle h,r,t\rangle\in\Delta\\ \langle h',r,t'\rangle\in\Delta'}} [\gamma + f_r(h,t) - f_r(h',t')]_+}^{\textsc{TransE}\ \textit{loss function}} + \sum_{\substack{\langle t,q,h\rangle\in\Delta_{\text{inverseOf}}\\ \langle t',q,h'\rangle\in\Delta'_{\text{inverseOf}}}} [\gamma + f_q(t,h) - f_q(t',h')]_+ \\
& + \sum_{\substack{\langle h,s,t\rangle\in\Delta_{\text{equivProperty}}\\ \langle h',s,t'\rangle\in\Delta'_{\text{equivProperty}}}} [\gamma + f_s(h,t) - f_s(h',t')]_+ + \sum_{\substack{\langle h,\text{typeOf},l\rangle\in\Delta\cup\in\Delta_{\text{equivClass}}\\ \langle h',\text{typeOf},l'\rangle\in\Delta'\cup\Delta'_{\text{equivClass}}}} [\gamma + f_{\text{typeOf}}(h,l) - f_{\text{typeOf}}(h',l')]_+ \\
& + \sum_{\substack{\langle h,\text{subClassOf},p\rangle\in\Delta_{\text{subClass}}\\ \langle h',\text{subClassOf},p'\rangle\in\Delta'_{\text{subClass}}}} [(\gamma-\beta) + f(h,p) - f(h',p')]_+
\end{aligned}
$$

where $q \equiv r^-$, $s \equiv r$ (properties), $l \equiv t$ and $t \sqsubseteq p$ (classes) and $f(h,p) = \|\mathrm{e}_h - \mathrm{e}_p\|$

TRANSROWL...

TRANSROWL

- adopts the same approach of TRANSOWL
- *is derived from* TRANSR [20]

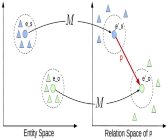
TRANSE $\Rightarrow$ poor modeling *reflexive* and *non* 1-to-1 relations (e.g. typeOf)

TRANSR $\Rightarrow$ more suitable to handle such specificity

TRANSR adopts TRANSE *loss function*

Score function

preliminarily projects e_s and e_o to the different d-dimensional space of the relational embeddings e_p through a suitable matrix $\mathrm{M} \in \mathbb{R}^{k \times d}$:



$$f'_p(\mathrm{e}_s, \mathrm{e}_o) = -\|(\mathrm{M}\mathrm{e}_s + \mathrm{e}_p) - \mathrm{M}\mathrm{e}_o\|_{\{1,2\}}.$$

where $\mathrm{e}'_s = \mathrm{M}\mathrm{e}_s$ and $\mathrm{e}'_o = \mathrm{M}\mathrm{e}_o$

[20] Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: AAAI 2015 Proceedings. (2015)

...TransROWL

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

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

where

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

TransROWL^R...

TransROWL^R adopts axiom-based regularization of *the loss function*, as for TransE^R [21]

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

TransE^R adopt TransE *score* and *loss function*
adds to the loss function *axiom-based regularizers* for inverse and equivalent property constraints

Loss function

$$L = \sum_{\substack{\langle h,r,t\rangle \in \Delta \\ (h',r',t') \in \Delta'}} [\gamma + f_r(h,t) - f_r(h',t')]_+ + \lambda \sum_{r \equiv q^- \in \mathcal{T}_{\text{inverseOf}}} \|r + q\| + \lambda \sum_{r \equiv p \in \mathcal{T}_{\text{equivProp}}} \|r - p\|$$

where $\mathcal{T}_{\text{inverseOf}}$ $\mathcal{T}_{\text{equivProp}}$ set of inverse properties and equivalent properties

[21] P. Minervini, L. Costabello, E. Muñoz, V. Novácek, P. Vandenbussche: Regularizing knowledge graph embeddings via equivalence and inversion axioms. ECML PKDD Proc. LNAI, vol. 10534, pp. 668–683 (2017)

...TransROWL^R

- TransR score function adopted
- *additional regularizers needed* for `equivalentClass` and `subClassOf` axioms
- *further constraints on the projection matrices* associated to relations

Loss function

$$
\begin{aligned}
L \;=\; & \sum_{\substack{\langle h,r,t\rangle \in \Delta \\ \langle h',r',t'\rangle \in \Delta'}} [\gamma + f'_r(h,t) - f'_r(h',t')]_+ \\
& + \lambda_1 \sum_{r \equiv q^- \in \mathcal{T}_{\text{inverseOf}}} \|r + q\| \; + \lambda_2 \sum_{r \equiv q^- \in \mathcal{T}_{\text{inverseOf}}} \|M_r - M_q\| \\
& + \lambda_3 \sum_{r \equiv p \in \mathcal{T}_{\text{equivProp}}} \|r - p\| \; + \lambda_4 \sum_{r \equiv p \in \mathcal{T}_{\text{equivProp}}} \|M_r - M_p\| \\
& + \lambda_5 \sum_{e' \equiv e'' \in \mathcal{T}_{\text{equivClass}}} \|e' - e''\| \; + \lambda_6 \sum_{s' \subseteq s'' \in \mathcal{T}_{\text{subClass}}} \|1 - \beta - (s' - s'')\|
\end{aligned}
$$

Additional term for projection matrices required for `inverseOf` and `equivProp` triples to favor the equality of their projection matrices

Lesson Learnt from Experiments...

Goal: Assessing the benefit of exploiting BK

- Comparing[22] TransOWL, TransROWL, TransROWL^R over to the original models TransE and TransR as a baseline

Perfomances tested on:

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

KGs adopted:

KG	*#Triples*	*#Entities*	*#Relationships*
DBpedia15K	180000	12800	278
DBpedia100K	600000	100000	321
DBpediaYAGO	290000	88000	316
NELL[23]	150000	68000	272

[22] All methods implemented as publicly available systems `https://github.com/Keehl-Mihael/TransROWL-HRS`

[23] 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

As for NELL, the models showed oscillating performances wrt the baselines

- NELL was aimed at testing in condition of larger incompleteness
  - `equivalentClass` and `equivalentProperty` **missing**
  - low number of typeOf-triples per entity

Conclusions

Conclusions:

- Symbol-based learning methods necessary for supplementing schema level information
- Exploiting BK to learn embeddings models may improve link prediction and triple classification results
- Deductive reasoning essential for the full usage of BK

Further Research Directions:

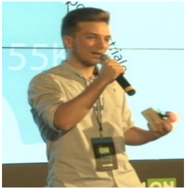
- Scalability of symbol-based learning methods still needs to be improved
- More robust enhanced KG embedding solutions needed for tackling KG incompleteness (case of NELL)
- Empower KG embedding methods with explanation tools
- Integrate further reasoning approaches (e.g. common sense reasoning)

Thank you

Nicola Fanizzi

Giuseppe Rizzo

Nicola F. Quatraro



Giovanni Sansaro

Formal Definition of Medoid

Medoid (notion drawn from the PAM algorithm)

- **central element in a group of instances**

$$m = \mathrm{medoid}(C) = \operatorname*{argmin}_{a \in C} \sum_{j=1}^{n} d(a, a_j)$$

- In presence of *outliers*, *medoids* are *more robust* than centroids (that are weighted average of points in a cluster)
  - The medoid is dictated by the location of predominant fraction of points inside a cluster

Distance measure between individuals adopted for TCT

Distance Function (adapted from [d'Amato et al.@ESWC2008]):

$$d_n^{\mathcal{C}} : \mathsf{Ind}(\mathcal{A}) \times \mathsf{Ind}(\mathcal{A}) \to [0, 1]$$

$$d_n^{\mathcal{C}}(a, b) = \left[\sum_{i=1}^{m} w_i \left[1 - \pi_i(a)\pi_i(b)\right]^n \right]^{1/n}$$

Context: a set of atomic concepts $\mathcal{C} = \{B_1, B_2, \ldots, B_m\}$

Projection Function:

$$\forall\, a \in \mathsf{Ind}(\mathcal{A}) \qquad \pi_i(a) = \begin{cases} 1 & \text{if } \mathcal{K} \models B_i(a) \\ 0 & \text{if } \mathcal{K} \models \neg B_i(a) \\ 0.5 & \text{otherwise} \end{cases}$$

Refinement Operators

Downward refinement operators specializing a concept C

1. $C' = C \sqcap (\neg)A$;
2. $C' = C \sqcap (\neg)(\exists)R.\top$;
3. $C' = C \sqcap (\neg)(\forall)R.\top$;
4. $\exists R.C'_i \in \rho(\exists R.C_i) \wedge C'_i \in \rho(C_i)$;
5. $\forall R.C'_i \in \rho(\forall R.C_i) \wedge C'_i \in \rho(C_i)$.

Ontologies per Experiments on TCT

Ontology	*DL Language*	*#Concepts*	*#Roles*	*#Individuals*	*#Disj. Ax.s*
BioPax	$\mathcal{ALCIF}(\mathcal{D})$	74	70	323	85
NTN	$\mathcal{SHIF}(\mathcal{D})$	47	27	676	40
Financial	$\mathcal{ALCIF}(\mathcal{D})$	60	16	1000	113
GeoSkills	$\mathcal{ALCHOIN}(\mathcal{D})$	596	23	2567	378
Monetary	$\mathcal{ALCHIF}(\mathcal{D})$	323	247	2466	236
DBPedia3.9	$\mathcal{ALCHI}(\mathcal{D})$	251	132	16606	11