

Machine learning with ontologies

Robert Hohendorf

Before the tutorial

See <https://github.com/bio-ontology-research-group/ontology-tutorial>:

- install Docker (e.g.: `apt-get install docker`)
- `docker pull coolmaksat/embeddings:latest`
- `docker pull leechuck/ontology-ml:latest`
- `docker run -i -t -p 8888:8888
leechuck/ontology-ml /bin/bash -c "jupyter
notebook
--notebook-dir=/home/borg/ontology-tutorial/
--ip='0.0.0.0' --port=8888 --no-browser
--allow-root"`

Overview

Ontologies and graphs

Machine learning and ontologies

- Graph-based methods

- Translating embeddings

- Syntactic approaches

- Model-theoretic approaches

Conclusion

Ontologies, axioms, and bioinformatics

- ontologies are ubiquitous
- rich formal characterization (axioms)
- how can they be used for (predictive) data analysis?
 - ▶ “fuzzy”, similarity-based search
 - ▶ predictive analysis and machine learning
 - ▶ background knowledge

Learning goals

- machine learning with ontologies as *features* (or background knowledge)
- unsupervised or supervised:
 - ▶ here: mostly unsupervised *feature* learning
- focus on existing tools and methods
 - ▶ Jupyter Notebooks and code examples
- not covered:
 - ▶ learning ontologies (axioms, definitions) from data
 - ▶ natural language processing
 - ▶ reasoning with ontologies
 - ▶ learning on “knowledge graphs”
 - ▶ machine learning theory

Agenda

- Introduction: ontologies, axioms, reasoning, graphs
- Machine learning:
 - ▶ graph-based
 - ▶ syntactic
 - ▶ model-theoretic
- (Semantic similarity)
 - ▶ probably not

Preliminaries: ontologies

- Specific artifacts expressing the intended meaning of a vocabulary in terms of primitive categories and relations describing the nature and structure of a domain of discourse
 - ▶ in order to account for the competent use of vocabulary in real situations (such as annotations in databases, etc.)
- the intended meaning of *primitive* categories and relations is expressed through axioms (axiomatic method, Tarski)

Preliminaries: axioms

- *classes* represent kinds of things in the world
 - ▶ *Arm, Apoptosis, Influenza, Homo sapiens, Drinking behavior, Membrane*
- *instances* of classes are individuals satisfying the classes' intension
 - ▶ my arm, the influenza I had last year, one ethanol molecule, etc.
- *relations* between instances arise from interactions, configurations, etc., of individuals
 - ▶ my arm is **part of** me, the **duration of** my influenza was 10 days
- *axioms* specify the conditions that instances of a class must satisfy
 - ▶ every instance of *Hand* is a **part of** an instance of *Arm*

Description Logics: overview

- TBox: axioms pertaining to the terminology of the domain (classes)
- ABox: axioms stating facts (assertions) about the world
- RBox: axioms holding for relations
- Reasoning: derive implicitly represented knowledge (e.g., subsumption)
- a “knowledge graph” is an ABox + RBox

Manchester OWL Syntax

DL Syntax	Manchester Syntax	Example
$C \sqcap D$	C and D	Human and Male
$C \sqcup D$	C or D	Male or Female
$\neg C$	not C	not Male
$\exists R.C$	R some C	hasChild some Human
$\forall R.C$	R only C	hasChild only Human
$(\geq nR.C)$	R min n C	hasChild min 1 Human
$(\leq nR.C)$	R max n C	hasChild max 1 Human
$(= nR.C)$	R exactly n C	hasChild exactly 1 Human
$\{a\} \sqcup \{b\} \sqcup \dots$	{a b ...}	{John Robert Mary}

Ontologies provide background knowledge

Annotation	Value
label	T cell aggregation
definition	The adhesion of one T cell to one or more other T cells via adhesion molecules.
class	http://purl.obolibrary.org/obo/GO_0070489
ontology	GO-PLUS
Equivalent	leukocyte aggregation and (has participant some T cell)
SubClassOf	lymphocyte aggregation , has participant some T cell
has_obo_namespace	biological_process
id	GO:0070489
synonyms	T-cell aggregation, T lymphocyte aggregation, T-lymphocyte aggregation

Ontologies provide background knowledge

Annotation	Value
label	T cell activation
definition	The change in morphology and behavior of a mature or immature T cell resulting from exposure to a mitogen, cytokine, chemokine, cellular ligand, or an antigen for which it is specific.
class	http://purl.obolibrary.org/obo/GO_0042110
ontology	GO-PLUS
Equivalent	cell activation and (has input some T cell)
SubClassOf	has input some T cell , lymphocyte activation
has_obo_namespace	biological_process
id	GO:0042110
synonyms	T-lymphocyte activation, T lymphocyte activation, T-cell activation

Ontologies provide background knowledge

<input type="checkbox"/>	MSN	Moesin	extracellular exosome	UniProt	Homo sapiens	HDA	
<input type="checkbox"/>	MSN	Moesin	T cell aggregation	UniProt	Homo sapiens	IDA	
<input type="checkbox"/>	MSN	Moesin	cellular response to testosterone stimulus	occurs in endothelial cell	UniProt	Homo sapiens	IDA
IL6	Interleukin-6	positive regulation of T cell proliferation	BHF-UCL	Homo sapiens	IDA		
IL6	Interleukin-6	T-helper 17 cell lineage commitment	UniProt	Homo sapiens	ISS		

Using background knowledge

Problem statement (first attempt):

Given a set of biological entities and their ontology-based annotations. Can we discover *new* relations between the biological entities, or between entities and classes in the ontology?

Using background knowledge

Problem statement (first attempt):

Given a set of biological entities and their ontology-based annotations. Can we discover *new* relations between the biological entities, or between entities and classes in the ontology?

- what relations, and when is a fact “new”?

Using background knowledge

Problem statement (first attempt):

Given a set of biological entities and their ontology-based annotations. Can we discover *new* relations between the biological entities, or between entities and classes in the ontology?

- what relations, and when is a fact “new”?
- what features are relevant?
 - ▶ depends on the relation!

Using background knowledge

Problem statement (first attempt):

Given a set of biological entities and their ontology-based annotations. Can we discover *new* relations between the biological entities, or between entities and classes in the ontology?

- what relations, and when is a fact “new”?
- what features are relevant?
 - ▶ depends on the relation!
- finding new facts is only one (minor?) use case
 - ▶ other uses: encode background knowledge for machine learning models (e.g., function prediction); add new classes; expand definition; etc.

Semantic similarity

semantic similarity measures:

- for words, terms, classes
- role of background knowledge:
 - ▶ statistical/distributional semantics, large corpora
 - ▶ ontologies: (graph) topology
- similarity measures: hand-crafted or data-driven?

Semantic similarity: some examples

- Are cyclin dependent kinases *functionally* more similar to lipid kinases or to riboflavin kinases? How about *phenotypically*?
- Which protein in the *mouse* is functionally most similar to the zebrafish *gustducin* protein?
- Which mouse knockout resembles *Bardet-Biedl Syndrome 8*?
- Are there mouse knockouts that resemble the side effects of diclofenac?
- Which genetic disease produces similar symptoms to ebola?
- Does functional similarity correlate with phenotypic similarity?

Semantic similarity or machine learning

- semantic similarity measures are mostly hand-crafted
 - ▶ capture certain intuition about what constitutes “similarity”
 - ▶ different measures for different kinds of similarity
 - ▶ usually interpretable (and explainable)

Semantic similarity or machine learning

- semantic similarity measures are mostly hand-crafted
 - ▶ capture certain intuition about what constitutes “similarity”
 - ▶ different measures for different kinds of similarity
 - ▶ usually interpretable (and explainable)
- machine learning methods are mostly data-driven
 - ▶ the architecture of the model is still hand-crafted
 - ▶ usually hard to interpret

Ontologies and graphs

- semantic similarity measures and machine learning models on ontologies can be graph-based, feature-based, or model-based

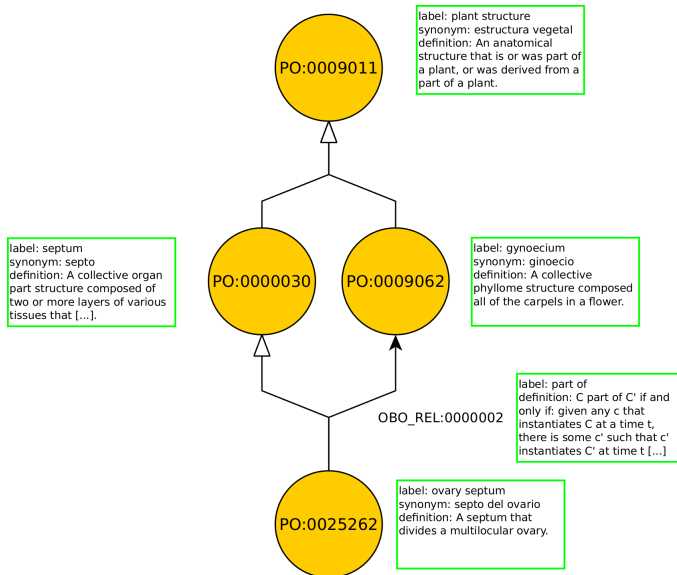
Ontologies and graphs

- semantic similarity measures and machine learning models on ontologies can be graph-based, feature-based, or model-based
- we may need to generate graphs from ontologies
 - ▶ *is-a* relations are easy (this is just `owl:subClassOf`)
 - ▶ how about *part-of*, *regulates*, *precedes*, etc.?
 - ▶ disjointness, universal vs. existential quantification, cardinality restrictions, intersection, union, negation?

Ontologies and graphs

- semantic similarity measures and machine learning models on ontologies can be graph-based, feature-based, or model-based
- we may need to generate graphs from ontologies
 - ▶ *is-a* relations are easy (this is just `owl:subClassOf`)
 - ▶ how about *part-of*, *regulates*, *precedes*, etc.?
 - ▶ disjointness, universal vs. existential quantification, cardinality restrictions, intersection, union, negation?
- relational patterns are implicit in OWL axioms
 - ▶ in first order logic
 - ▶ needs to translate them into OWL
 - ▶ defined in OBO Relation Ontology

Relations as patterns



Relations as patterns

- OBO Relation Ontology (RO):
 - ▶ <https://github.com/oborel/obo-relations>
- Basic Formal Ontology (BFO):
 - ▶ provides top-level classes
 - ▶ Continuant, Process, Function, Material object, etc.
 - ▶ used for some OBO Foundry ontologies
- RO and BFO provide a top-level system of classes and relations shared across many biomedical ontologies
- this system may define patterns used to generate graphs

Relations as patterns

- X SubClassOf: $Y: X \xrightarrow{\text{is-a}} Y$
- X SubClassOf: part-of some Y: $X \xrightarrow{\text{part-of}} Y$
- X SubClassOf: regulates some Y: $X \xrightarrow{\text{regulates}} Y$
- X DisjointWith: $Y: X \xleftrightarrow{\text{disjoint}} Y$
- X EquivalentTo: $Y: X \xleftrightarrow{=} Y, \{X, Y\}$

Asserted and inferred

- relation patterns can be asserted or inferred
- `X SubClassOf: part-of some Y`
- `Y SubClassOf: part-of some Z`
- `part-of o part-of SubPropertyOf: part-of`
- \vdash `X SubClassOf: part-of some Z`

OBO Format represents ontologies as graphs:

- Protege/OWLAPI: OBO export
- OBO toolsets (e.g., ROBOT)
- [https://github.com/bio-ontology-research-group/](https://github.com/bio-ontology-research-group/Onto2Graph)
Onto2Graph

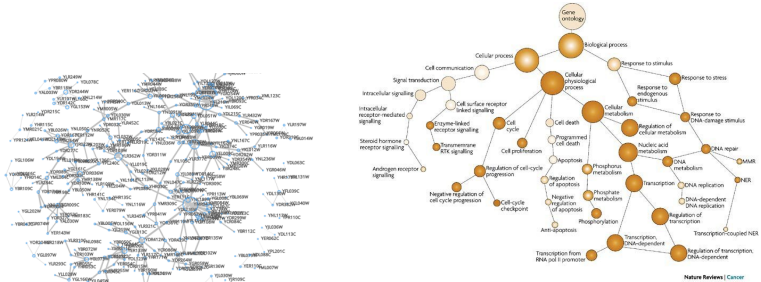
OBO Format represents ontologies as graphs:

- Protege/OWLAPI: OBO export
- OBO toolsets (e.g., ROBOT)
- [https://github.com/bio-ontology-research-group/](https://github.com/bio-ontology-research-group/Onto2Graph)
Onto2Graph
- a conversion of an ontologies into a graph will almost always lead to a loss of information

OBO Format represents ontologies as graphs:

- Protege/OWLAPI: OBO export
- OBO toolsets (e.g., ROBOT)
- <https://github.com/bio-ontology-research-group/On2Graph>
- a conversion of an ontologies into a graph will almost always lead to a loss of information
- edges should be “meaningful”: not merely syntax
 - ▶ the RDF serialization of OWL is a graph and contains all information but is a bad idea for semantic similarity or machine learning (more later)

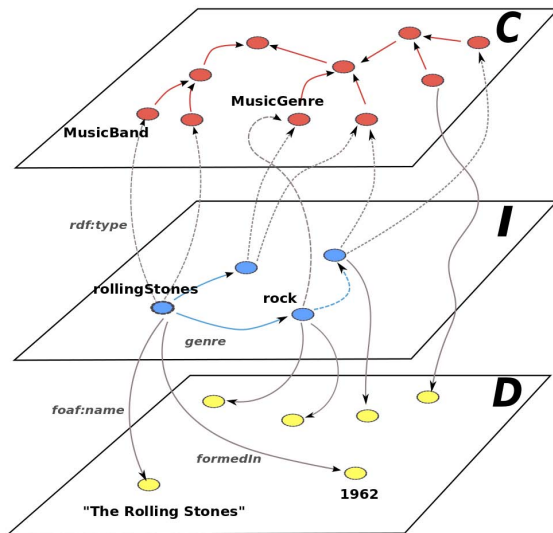
An example: protein–protein interactions and GO functions



Machine learning with ontologies: approaches

- graph-based
- syntactic
- model-theoretic

How to measure similarity?



How to measure similarity?

- Shortest Path

- ▶ applicable to arbitrary “knowledge graphs”
- ▶ does not capture similarity well over all edge types, e.g., *disjointWith*, *differentFrom*, *opposite-of*, etc.

- Random Walk

- ▶ with or without restart
- ▶ iterated
- ▶ does not consider edge labels \Rightarrow captures only adjacency of nodes
- ▶ scores whole graph with *probability* of being in a state
- ▶ can take multiple seed nodes
 - ▶ can be used to find disease genes

Graph-based learning

- feature learning on graphs

Graph-based learning

- feature learning on graphs
- e.g., iterated, edge-labeled random walk
 - ▶ walks form *sentences*
 - ▶ sentences form a *corpus*
 - ▶ feature learning on corpus through Word2Vec (or factorization of co-occurrence matrix)
 - ▶ RDF2Vec: <http://data.dws.informatik.uni-mannheim.de/rdf2vec/>
 - ▶ with support for reasoning over ontologies:
<https://github.com/bio-ontology-research-group/walking-rdf-and-owl>

Graph-based learning

- feature learning on graphs
- e.g., iterated, edge-labeled random walk
 - ▶ walks form *sentences*
 - ▶ sentences form a *corpus*
 - ▶ feature learning on corpus through Word2Vec (or factorization of co-occurrence matrix)
 - ▶ RDF2Vec: <http://data.dws.informatik.uni-mannheim.de/rdf2vec/>
 - ▶ with support for reasoning over ontologies:
<https://github.com/bio-ontology-research-group/walking-rdf-and-owl>
- Translational knowledge graph embeddings: TransE, TransR, TransE, HolE, etc.
 - ▶ analogy- or translation-based
 - ▶ <https://github.com/SmartDataAnalytics/PyKEEN>

Graph-based learning

- feature learning on graphs
- e.g., iterated, edge-labeled random walk
 - ▶ walks form *sentences*
 - ▶ sentences form a *corpus*
 - ▶ feature learning on corpus through Word2Vec (or factorization of co-occurrence matrix)
 - ▶ RDF2Vec: <http://data.dws.informatik.uni-mannheim.de/rdf2vec/>
 - ▶ with support for reasoning over ontologies:
<https://github.com/bio-ontology-research-group/walking-rdf-and-owl>
- Translational knowledge graph embeddings: TransE, TransR, TransE, HolE, etc.
 - ▶ analogy- or translation-based
 - ▶ <https://github.com/SmartDataAnalytics/PyKEEN>
- Graph Convolution Neural Networks (not discussed here)

Graph embeddings

Definition

Let $KG = (V, E, L; \vdash)$ be an ontology graph with a set of vertices V , a set of edges $E \subseteq V \times V$, a label function $L : V \cup E \mapsto Lab$ that assigns labels from a set of labels Lab to vertices and edges, and an inference relation \vdash . An ontology graph embedding is a function $f_\eta : L(V) \cup L(E) \mapsto \mathbf{R}^n$.

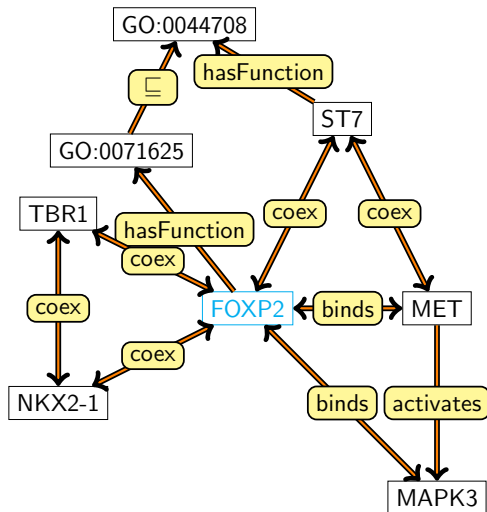
Graph embeddings

Definition

Let $KG = (V, E, L; \vdash)$ be an ontology graph with a set of vertices V , a set of edges $E \subseteq V \times V$, a label function $L : V \cup E \mapsto Lab$ that assigns labels from a set of labels Lab to vertices and edges, and an inference relation \vdash . An ontology graph embedding is a function $f_\eta : L(V) \cup L(E) \mapsto \mathbb{R}^n$.

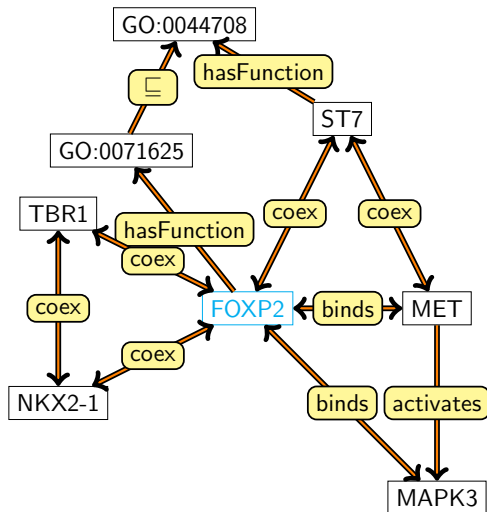
- key idea: preserve *some* structure of the graph in \mathbb{R}^n (under operations in \mathbb{R}^n)
- \mathbb{R}^n enables *new* operations (such as many similarity measures)
- useful as *feature* vectors

Random walks



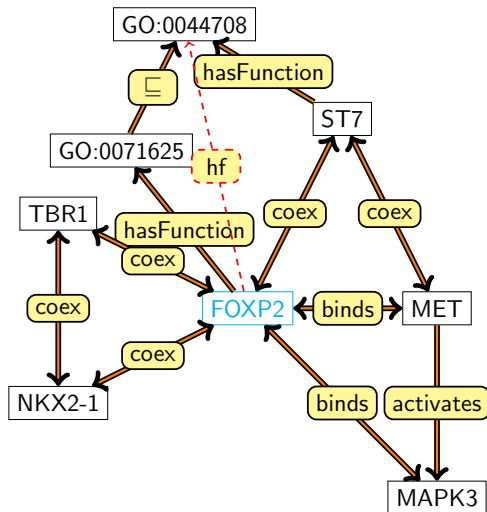
- FOXP2 is characterized by *adjacent* and close nodes and edges
- different edges may “transmit” information differently

Random walks



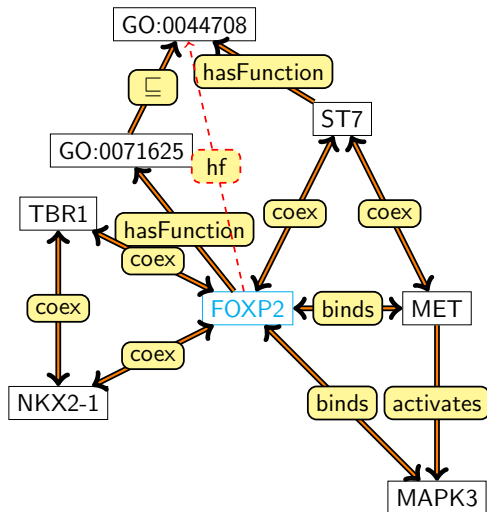
- precompute the deductive closure:
- for all ϕ : if $\mathcal{KG} \models \phi$, add ϕ to \mathcal{KG}

Random walks



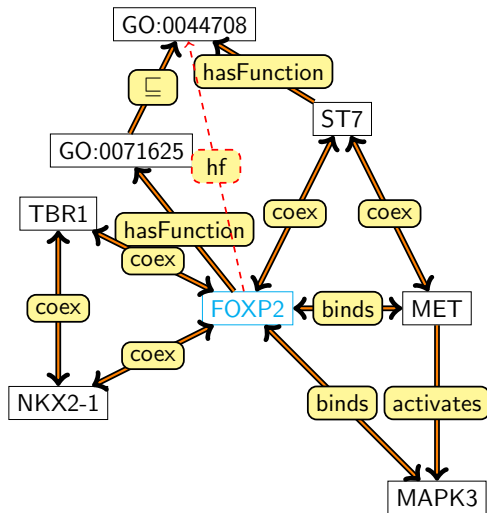
- precompute the deductive closure:
- for all ϕ : if $\mathcal{KG} \models \phi$, add ϕ to \mathcal{KG}

Random walks



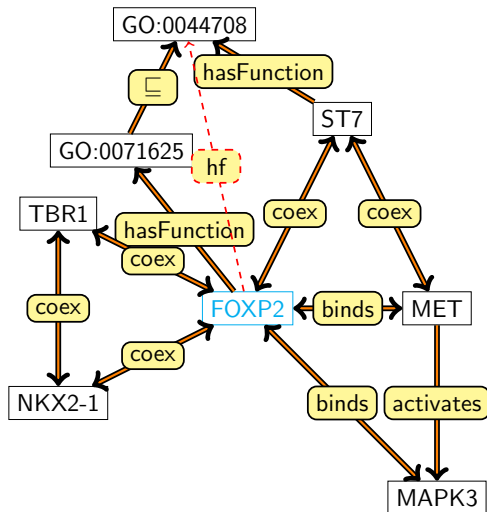
- Exploring the graph:

Random walks



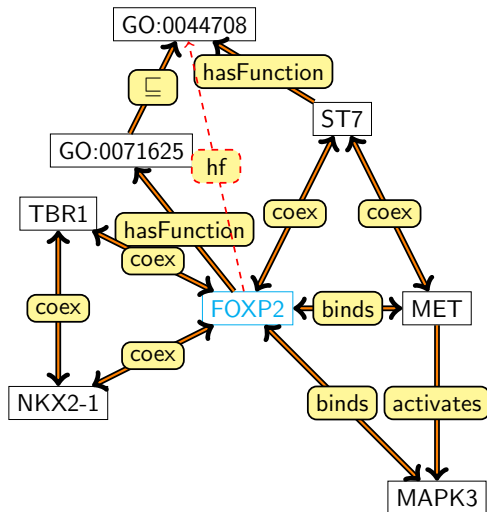
- Exploring the graph:
- :FOXP2 :binds :MET
:coex :ST7
:hasFunction
GO:0044708

Random walks



- Exploring the graph:
- :FOXP2 :binds :MET
:coex :ST7
:hasFunction
GO:0044708
- :FOXP2 :hasFunction
GO:0071625
subClassOf
GO:0044708

Random walks



- Exploring the graph:
- `:FOXP2 :binds :MET`
`:coex :ST7`
`:hasFunction`
`GO:0044708`
- `:FOXP2 :hasFunction`
`GO:0071625`
`subClassOf`
`GO:0044708`
- `:FOXP2 :coex :TBR1`
`:coex :NKX2-1`
`:coex`
`:TBR1 :coex ...`

Maximize:

$$\frac{1}{N} \sum_{n=1}^N \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{n+j} | w_n) \quad (1)$$

with

$$p(w_o | w_i) = \frac{\exp(v'_{w_o}{}^T v_{w_i})}{\sum_{w=1}^W \exp(v'_w{}^T v_{w_i})} \quad (2)$$

(at least conceptually; different strategies are used to approximate Eqn. 2)

Word2Vec

Source Text

Training Samples

The quick brown fox jumps over the lazy dog. →	(the, quick) (the, brown)
The quick brown fox jumps over the lazy dog. →	(quick, the) (quick, brown) (quick, fox)
The quick brown fox jumps over the lazy dog. →	(brown, the) (brown, quick) (brown, fox) (brown, jumps)
The quick brown fox jumps over the lazy dog. →	(fox, quick) (fox, brown) (fox, jumps) (fox, over)

Word2Vec and Random Walks

- random walks “flatten” a graph
 - ▶ walks capture node neighborhood
 - ▶ and generate a “corpus”
- random walks capture graph “structure”
 - ▶ hub-nodes, communities, etc.
 - ▶ determine “importance” of nodes
- embeddings capture co-occurrence
 - ▶ similar graph neighborhood \Rightarrow similar co-occurrence \Rightarrow similar vector
- embeddings generate “feature” vectors
 - ▶ functions from symbols (words, labels) into \mathbb{R}^n

What to do with embeddings?

- useful for edge prediction, similarity, clustering, as feature vectors

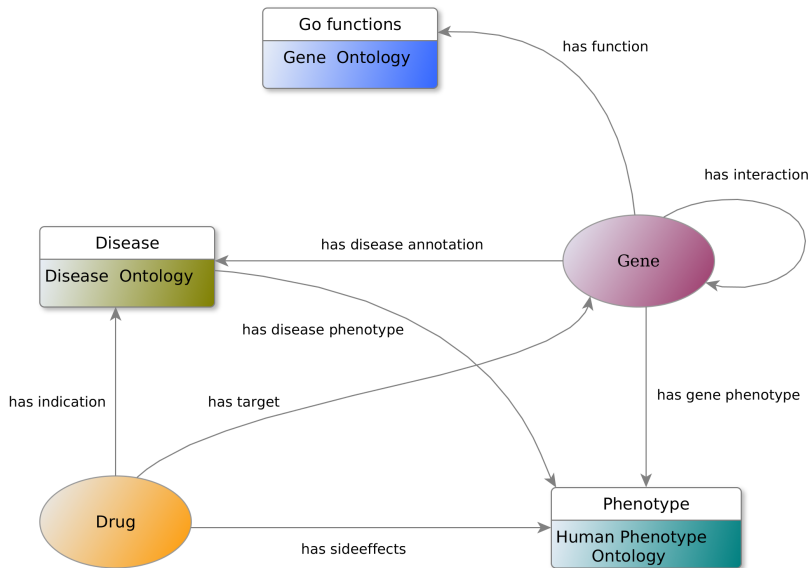
- ▶ supervised: edge prediction (e.g., SVM, ANN)

- ▶ e.g.: find a function $f : \mathbb{R}^n \times \mathbb{R}^n \mapsto [0, 1]$ s.t. $\sqrt{\frac{\sum_{t=1}^T (\hat{y}_t - y_t)^2}{T}}$ (RMSE) is minimized for a set of true labels y_k

- ▶ unsupervised: clustering, similarity, visualization

- ▶ cosine similarity (for L2-normalized features)
- ▶ Word2Vec embeddings capture similarity between co-occurrence vectors

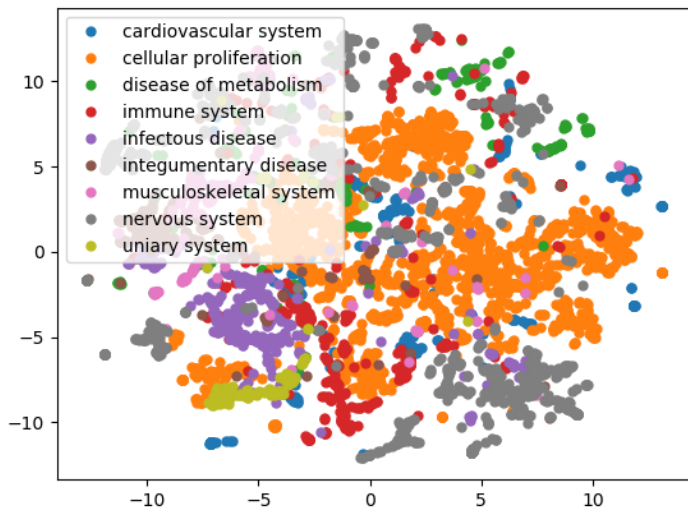
Random walks



Visualizing feature vectors: dimensionality reduction

- project n -dimensional vectors in 2D (or 3D) space
- and color with some known labels
 - ▶ high-level/general classes in an ontology work great
- PCA or t-SNE
- <https://lvdmaaten.github.io/tsne/>

Visualizing feature vectors



Features: supervised learning

- feature vectors represent graph neighborhood of nodes
 - ▶ adjacent nodes and edges
 - ▶ ontology classes (asserted & inferred)
- useful in supervised prediction tasks
- relation prediction:
 - ▶ input: two features vectors (from embedding function)
 - ▶ output: 0 or 1 (relation or not)
 - ▶ training data: positive and negative cases
 - ▶ $R(x, y)$ and $\neg R(x, y)$

Features: supervised learning

Object property	Source type	Target type	Without reasoning		With reasoning	
			F-measure	AUC	F-measure	AUC
has target	Drug	Gene/Protein	0.94	0.97	0.94	0.98
has disease annotation	Gene/Protein	Disease	0.89	0.95	0.89	0.95
has side-effect*	Drug	Phenotype	0.86	0.93	0.87	0.94
has interaction	Gene/Protein	Gene/Protein	0.82	0.88	0.82	0.88
has function*	Gene/Protein	Function	0.85	0.95	0.83	0.91
has gene phenotype*	Gene/Protein	Phenotype	0.84	0.91	0.82	0.90
has indication	Drug	Disease	0.72	0.79	0.76	0.83
has disease phenotype*	Disease	Phenotype	0.72	0.78	0.70	0.77

The forkhead-box P2 (FOXP2) gene polymorphism has been reported to be involved in the susceptibility to schizophrenia; however, few studies have investigated the association between FOXP2 gene polymorphism and clinical symptoms in schizophrenia.

The forkhead-box P2 (FOXP2) gene polymorphism has been reported to be involved in the susceptibility to schizophrenia; however, few studies have investigated the association between FOXP2 gene polymorphism and clinical symptoms in schizophrenia.

- FOXP2 :binds :MET :coex :ST7 :hasFunction GO:0044708
- FOXP2 :hasFunction GO:0071625 subClassOf GO:0044708
- FOXP2 :coex :TBR1 :coex :NKX2-1 :coex :TBR1 :coex ...

The :FOXP2 gene polymorphism has been reported to be involved in the susceptibility to schizophrenia; however, few studies have investigated the association between :FOXP2 gene polymorphism and clinical symptoms in schizophrenia.

- :FOXP2 :binds :MET :coex :ST7 :hasFunction GO:0044708
- :FOXP2 :hasFunction GO:0071625 subClassOf GO:0044708
- :FOXP2 :coex :TBR1 :coex :NKX2-1 :coex :TBR1 :coex ...

- RDF2Vec: random walks on RDF + Word2Vec
- RDF2Vec: Weisfeiler-Lehmann kernel on RDF
- <https://datalab.rwth-aachen.de/embedding/RDF2Vec/>

- RDF2Vec: random walks on RDF + Word2Vec
- RDF2Vec: Weisfeiler-Lehmann kernel on RDF
- <https://datalab.rwth-aachen.de/embedding/RDF2Vec/>
- Walking RDF+OWL: random walks on RDF + Elk + Word2Vec
 - ▶ inference
- <https://github.com/bio-ontology-research-group/walking-rdf-and-owl>

Some limitations

- “word”-based (Word2Vec):
 - ▶ semantics is reduced to co-occurrence (in ABox/TBox statements)
 - ▶ “disjointWith” vs. “part-of” vs. “subClassOf”

Jupyter exercise

- Open the Jupyter notebook `graph.ipynb`
- Follow the examples in the first part of the notebook (random walks)
- If you don't have a powerful CPU in your laptop (with multiple cores), you may want to lower the number of iterations (`n_iter`) during TSNE
- some of the code will take a while to run
 - ▶ if things are too slow, you can keep it running while we continue or complete this after the tutorial

Translating embeddings

Definition

Let $KG = (V, E, L; \vdash)$ be a knowledge graph with a set of vertices V , a set of edges $E \subseteq V \times V$, a label function $L : V \cup E \mapsto Lab$ that assigns labels from a set of labels Lab to vertices and edges, and an inference relation \vdash . A knowledge graph embedding is a function $f_\eta : L(V) \cup L(E) \mapsto \mathbf{R}^n$.

Translating embeddings

Definition

Let $KG = (V, E, L; \vdash)$ be a knowledge graph with a set of vertices V , a set of edges $E \subseteq V \times V$, a label function $L : V \cup E \mapsto Lab$ that assigns labels from a set of labels Lab to vertices and edges, and an inference relation \vdash . A knowledge graph embedding is a function $f_\eta : L(V) \cup L(E) \mapsto \mathbf{R}^n$.

Graph as edgelist: set of (s, p, o) statements

Translating embeddings

Definition

Let $KG = (V, E, L; \vdash)$ be a knowledge graph with a set of vertices V , a set of edges $E \subseteq V \times V$, a label function $L : V \cup E \mapsto Lab$ that assigns labels from a set of labels Lab to vertices and edges, and an inference relation \vdash . A knowledge graph embedding is a function $f_\eta : L(V) \cup L(E) \mapsto \mathbf{R}^n$.

Graph as edgelist: set of (s, p, o) statements

Idea: $\mu(s) + \mu(p) \approx \mu(o)$

Translating embeddings

Definition

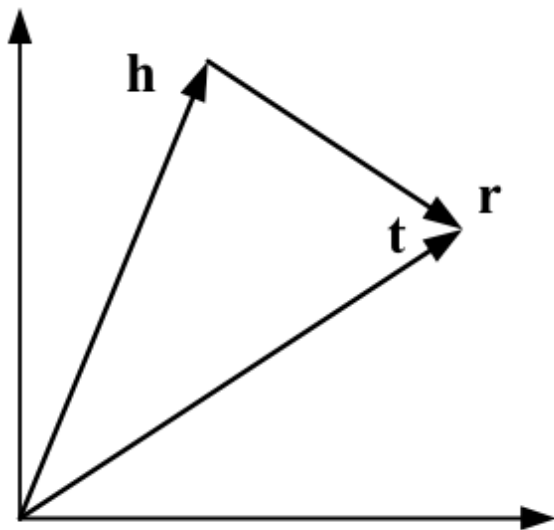
Let $KG = (V, E, L; \vdash)$ be a knowledge graph with a set of vertices V , a set of edges $E \subseteq V \times V$, a label function $L : V \cup E \mapsto Lab$ that assigns labels from a set of labels Lab to vertices and edges, and an inference relation \vdash . A knowledge graph embedding is a function $f_\eta : L(V) \cup L(E) \mapsto \mathbf{R}^n$.

Graph as edgelist: set of (s, p, o) statements

Idea: $\mu(s) + \mu(p) \approx \mu(o)$

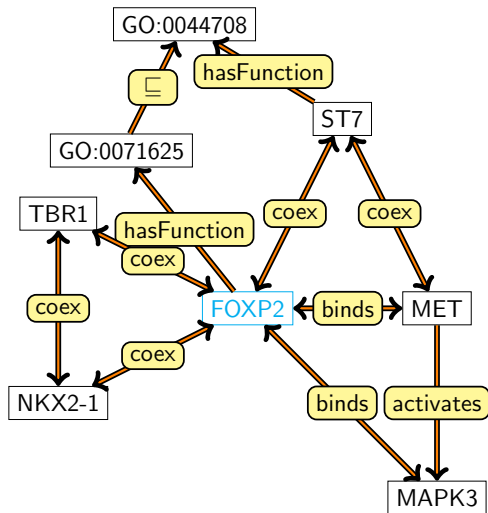
Minimize: $\sum_t \|\mu(s) + \mu(p) - \mu(o)\|$ (chose your norm, usually L2)

Translating embeddings

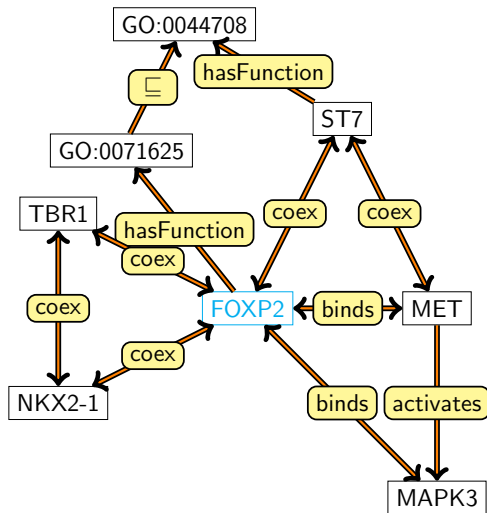


Entity and Relation Space

Translating embeddings

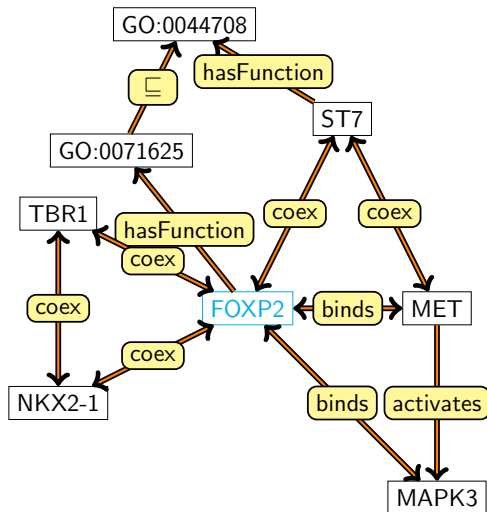


Translating embeddings



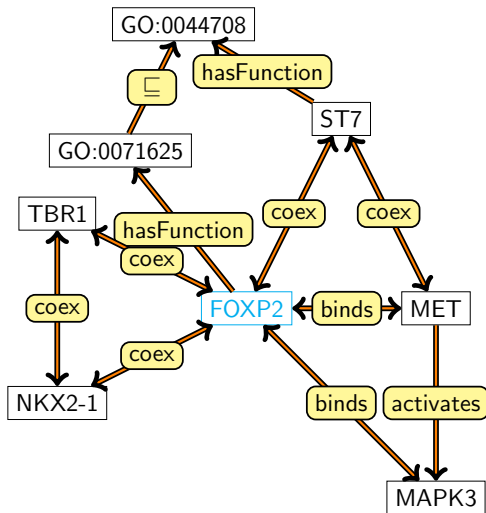
- $\text{FOXP2} + \text{binds} = \text{MET}$

Translating embeddings



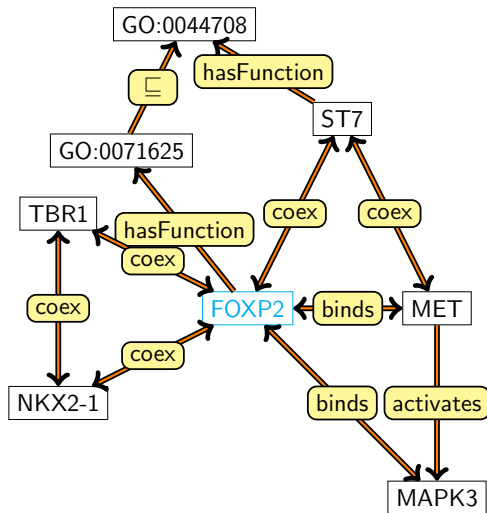
- FOXP2 + binds = MET
- MAP + activates = MAPK3

Translating embeddings



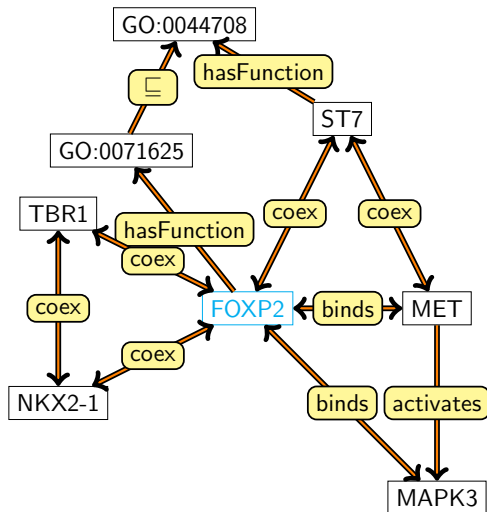
- FOXP2 + binds = MET
- MAP + activates = MAPK3
- MET + binds = FOXP2

Translating embeddings



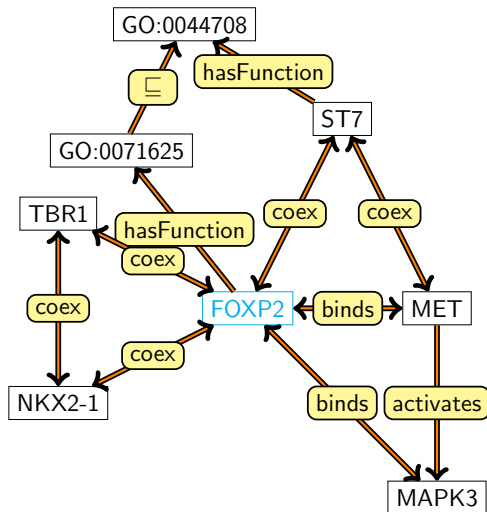
- FOXP2 + binds = MET
- MAP + activates = MAPK3
- MET + binds = FOXP2
- ST7 + hasFunction = G0:0044708

Translating embeddings



- FOXP2 + binds = MET
- MAP + activates = MAPK3
- MET + binds = FOXP2
- ST7 + hasFunction = GO:0044708
- ...

Translating embeddings



- $\text{FOXP2} + \text{binds} - \text{MET} = 0$
- $\text{MAP} + \text{activates} - \text{MAPK3} = 0$
- $\text{MET} + \text{binds} - \text{FOXP2} = 0$
- $\text{ST7} + \text{hasFunction} - \text{GO:0044708} = 0$
- ...

Translating embeddings

Algorithm 1 Learning TransE

input Training set $S = \{(h, \ell, t)\}$, entities and rel. sets E and L , margin γ , embeddings dim. k .

```
1: initialize  $\ell \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$  for each  $\ell \in L$ 
2:            $\ell \leftarrow \ell / \|\ell\|$  for each  $\ell \in L$ 
3:            $\mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$  for each entity  $e \in E$ 
4: loop
5:    $\mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$  for each entity  $e \in E$ 
6:    $S_{batch} \leftarrow \text{sample}(S, b)$  // sample a minibatch of size  $b$ 
7:    $T_{batch} \leftarrow \emptyset$  // initialize the set of pairs of triplets
8:   for  $(h, \ell, t) \in S_{batch}$  do
9:      $(h', \ell, t') \leftarrow \text{sample}(S'_{(h, \ell, t)})$  // sample a corrupted triplet
10:     $T_{batch} \leftarrow T_{batch} \cup \{((h, \ell, t), (h', \ell, t'))\}$ 
11:   end for
12:   Update embeddings w.r.t. 
$$\sum_{((h, \ell, t), (h', \ell, t')) \in T_{batch}} \nabla [\gamma + d(\mathbf{h} + \ell, \mathbf{t}) - d(\mathbf{h}' + \ell, \mathbf{t}')]_+$$

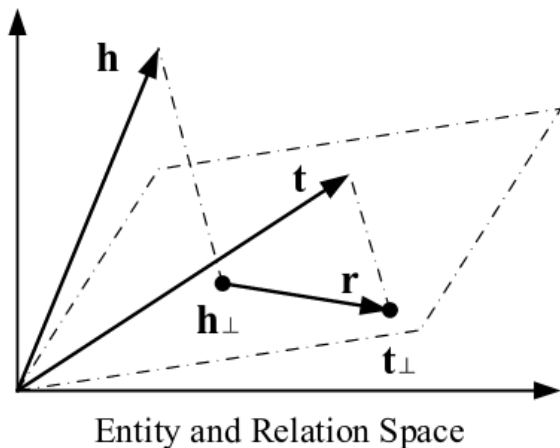
13: end loop
```

Bordes et al. (2013). Translating Embeddings for Modeling Multi-relational Data.

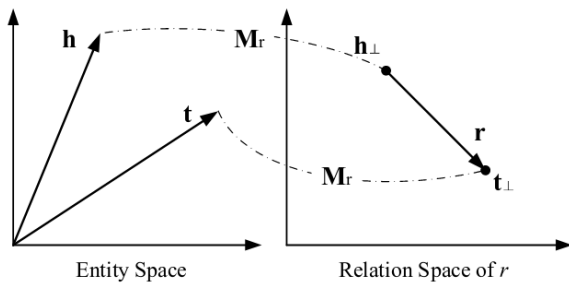
Some properties of TransE

- graph-based
 - ▶ works well on RDF graphs
 - ▶ and ontology graphs
- 1:1 relations only
 - ▶ not suitable for hierarchies (1-N relations)
 - ▶ not suitable for N-N relations
 - ▶ no transitive, symmetric, reflexive relations

Translating embeddings



Translating embeddings



(c) TransR.

Translating embeddings

Method	Ent. embedding	Rel. embedding	Scoring function $f_r(h, t)$	Constraints/Regularization
TransE [14]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^d$	$-\ \mathbf{h} + \mathbf{r} - \mathbf{t}\ _{1/2}$	$\ \mathbf{h}\ _2 = 1, \ \mathbf{t}\ _2 = 1$
TransH [15]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r}, \mathbf{w}_r \in \mathbb{R}^d$	$-\ (\mathbf{h} - \mathbf{w}_r^\top \mathbf{h} \mathbf{w}_r) + \mathbf{r} - (\mathbf{t} - \mathbf{w}_r^\top \mathbf{t} \mathbf{w}_r)\ _2^2$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1$ $\ \mathbf{w}_r^\top \mathbf{r}\ / \ \mathbf{r}\ _2 \leq c, \ \mathbf{w}_r\ _2 = 1$
TransR [16]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^k, \mathbf{M}_r \in \mathbb{R}^{k \times d}$	$-\ \mathbf{M}_r \mathbf{h} + \mathbf{r} - \mathbf{M}_r \mathbf{t}\ _2^2$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1, \ \mathbf{r}\ _2 \leq 1$ $\ \mathbf{M}_r \mathbf{h}\ _2 \leq 1, \ \mathbf{M}_r \mathbf{t}\ _2 \leq 1$
TransD [50]	$\mathbf{h}, \mathbf{w}_h \in \mathbb{R}^d$ $\mathbf{t}, \mathbf{w}_t \in \mathbb{R}^d$	$\mathbf{r}, \mathbf{w}_r \in \mathbb{R}^k$	$-\ (\mathbf{w}_r \mathbf{w}_h^\top + \mathbf{I})\mathbf{h} + \mathbf{r} - (\mathbf{w}_r \mathbf{w}_t^\top + \mathbf{I})\mathbf{t}\ _2^2$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1, \ \mathbf{r}\ _2 \leq 1$ $\ (\mathbf{w}_r \mathbf{w}_h^\top + \mathbf{I})\mathbf{h}\ _2 \leq 1$ $\ (\mathbf{w}_r \mathbf{w}_t^\top + \mathbf{I})\mathbf{t}\ _2 \leq 1$
TransSparse [51]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^k, \mathbf{M}_r(\theta_r) \in \mathbb{R}^{k \times d}$ $\mathbf{M}_r^1(\theta_r^1), \mathbf{M}_r^2(\theta_r^2) \in \mathbb{R}^{k \times d}$	$-\ \mathbf{M}_r(\theta_r)\mathbf{h} + \mathbf{r} - \mathbf{M}_r(\theta_r)\mathbf{t}\ _{1/2}^2$ $-\ \mathbf{M}_r^1(\theta_r^1)\mathbf{h} + \mathbf{r} - \mathbf{M}_r^2(\theta_r^2)\mathbf{t}\ _{1/2}^2$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1, \ \mathbf{r}\ _2 \leq 1$ $\ \mathbf{M}_r(\theta_r)\mathbf{h}\ _2 \leq 1, \ \mathbf{M}_r(\theta_r)\mathbf{t}\ _2 \leq 1$ $\ \mathbf{M}_r^1(\theta_r^1)\mathbf{h}\ _2 \leq 1, \ \mathbf{M}_r^2(\theta_r^2)\mathbf{t}\ _2 \leq 1$
TransM [52]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^d$	$-\theta_r \ \mathbf{h} + \mathbf{r} - \mathbf{t}\ _{1/2}$	$\ \mathbf{h}\ _2 = 1, \ \mathbf{t}\ _2 = 1$
ManifoldE [53]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^d$	$-(\ \mathbf{h} + \mathbf{r} - \mathbf{t}\ _2^2 - \theta_r^2)^2$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1, \ \mathbf{r}\ _2 \leq 1$
TransF [54]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^d$	$(\mathbf{h} + \mathbf{r})^\top \mathbf{t} + (\mathbf{t} - \mathbf{r})^\top \mathbf{h}$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1, \ \mathbf{r}\ _2 \leq 1$
TransA [55]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{r} \in \mathbb{R}^d, \mathbf{M}_r \in \mathbb{R}^{d \times d}$	$-(\mathbf{h} + \mathbf{r} - \mathbf{t})^\top \mathbf{M}_r (\mathbf{h} + \mathbf{r} - \mathbf{t})$	$\ \mathbf{h}\ _2 \leq 1, \ \mathbf{t}\ _2 \leq 1, \ \mathbf{r}\ _2 \leq 1$ $\ \mathbf{M}_r\ _F \leq 1, [\mathbf{M}_r]_{ij} = [\mathbf{M}_r]_{ji} \geq 0$
KG2E [45]	$\mathbf{h} \sim \mathcal{N}(\mu_h, \Sigma_h)$ $\mathbf{t} \sim \mathcal{N}(\mu_t, \Sigma_t)$ $\mu_h, \mu_t \in \mathbb{R}^d$ $\Sigma_h, \Sigma_t \in \mathbb{R}^{d \times d}$	$\mathbf{r} \sim \mathcal{N}(\mu_r, \Sigma_r)$ $\mu_r \in \mathbb{R}^d, \Sigma_r \in \mathbb{R}^{d \times d}$	$-\text{tr}(\Sigma_r^{-1}(\Sigma_h + \Sigma_t)) - \mu^\top \Sigma_r^{-1} \mu - \ln \frac{\det(\Sigma_r)}{\det(\Sigma_h + \Sigma_t)}$ $-\mu^\top \Sigma^{-1} \mu - \ln(\det(\Sigma))$ $\mu = \mu_h + \mu_r - \mu_t$ $\Sigma = \Sigma_h + \Sigma_r + \Sigma_t$	$\ \mu_h\ _2 \leq 1, \ \mu_t\ _2 \leq 1, \ \mu_r\ _2 \leq 1$ $c_{min} \mathbf{I} \leq \Sigma_h \leq c_{max} \mathbf{I}$ $c_{min} \mathbf{I} \leq \Sigma_t \leq c_{max} \mathbf{I}$ $c_{min} \mathbf{I} \leq \Sigma_r \leq c_{max} \mathbf{I}$
TransG [46]	$\mathbf{h} \sim \mathcal{N}(\mu_h, \sigma_h^2 \mathbf{I})$ $\mathbf{t} \sim \mathcal{N}(\mu_t, \sigma_t^2 \mathbf{I})$ $\mu_h, \mu_t \in \mathbb{R}^d$	$\mu_r \sim \mathcal{N}(\mu_r, (\sigma_r^2 + \sigma_t^2) \mathbf{I})$ $\mathbf{r} = \sum_i \pi_r \mu_r^i \in \mathbb{R}^d$	$\sum_i \pi_r \exp\left(-\frac{\ \mu_h + \mu_r^i - \mu_t\ _2^2}{\sigma_h^2 + \sigma_t^2}\right)$	$\ \mu_h\ _2 \leq 1, \ \mu_t\ _2 \leq 1, \ \mu_r^i\ _2 \leq 1$
UM [56]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	—	$-\ \mathbf{h} - \mathbf{t}\ _2^2$	$\ \mathbf{h}\ _2 = 1, \ \mathbf{t}\ _2 = 1$
SE [57]	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$	$\mathbf{M}_r^1, \mathbf{M}_r^2 \in \mathbb{R}^{d \times d}$	$-\ \mathbf{M}_r^1 \mathbf{h} - \mathbf{M}_r^2 \mathbf{t}\ _1$	$\ \mathbf{h}\ _2 = 1, \ \mathbf{t}\ _2 = 1$

Wang et al. Knowledge Graph Embedding: A Survey of Approaches and Applications.

- Python package to generate knowledge graph embeddings
- supports many different graph embedding types: TransE, TransR, TransD, RESCAL, etc.
- hyperparameter optimization (“HPO”) and evaluation included
- <https://github.com/SmartDataAnalytics/PyKEEN>

Some limitations

- graph-based (same as random walks):
 - ▶ ontologies are not graphs!
 - ▶ converting ontologies to graphs loses information
 - ▶ no axioms, no definitions

Jupyter exercise

- run the PyKEEN part of `graph.ipynb`
- again: this may take a while
- you can also explore
<https://github.com/SmartDataAnalytics/PyKEEN>
- try to expand the notebook to predict “new” relations
 - ▶ using numpy directly, or PyKEEN’s predictions methods

Ontologies: axioms, not graphs!

Overview

Browse

DLQuery

Download

Annotation	Value
label	B cell apoptotic process
definition	Any apoptotic process in a B cell, a lymphocyte of B lineage with the phenotype CD19-positive and capable of B cell mediated immunity.
class	http://purl.obolibrary.org/obo/GO_0001783
ontology	GO-PLUS
Equivalent	apoptotic process and (occurs in some B cell)
SubClassOf	occurs in some B cell , lymphocyte apoptotic process
id	GO:0001783
has_obo_namespace	biological_process

Ontologies: axioms, not graphs!

Gene Ontology:

- behavior DisjointWith: 'developmental process'
- behavior SubclassOf: only-in-taxon some metazoa
- 'cell proliferation' DisjointWith: in-taxon some fungi
- 'cell growth' EquivalentTo: growth and ('results in growth of' some cell)
- ...

Ontology embeddings

Definition

Let $O = (\Sigma = (C, R, I); ax; \vdash)$ be an ontology with a set of classes C , a set of relations R , a set of instances I , a set of axioms ax and an inference relation \vdash . An ontology embedding is a function $f_\eta : C \cup R \cup I \mapsto \mathbb{R}^n$ (or $\Sigma(O) \mapsto \mathbb{R}^n$).

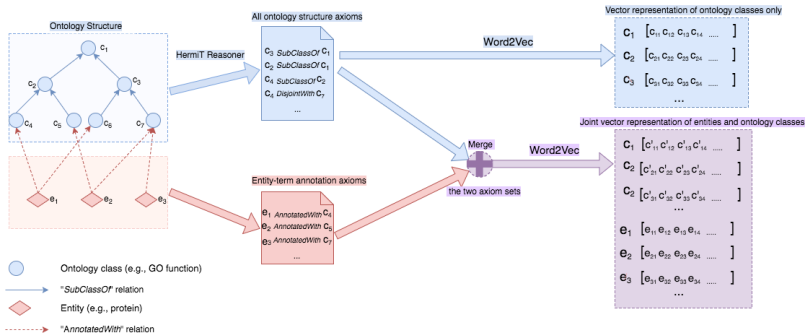
Ontology embeddings

Definition

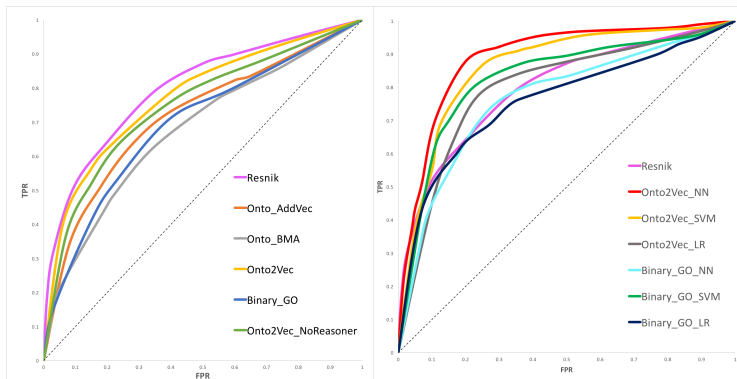
Let $O = (\Sigma = (C, R, I); ax; \vdash)$ be an ontology with a set of classes C , a set of relations R , a set of instances I , a set of axioms ax and an inference relation \vdash . An ontology embedding is a function $f_\eta : C \cup R \cup I \mapsto \mathbb{R}^n$ (or $\Sigma(O) \mapsto \mathbb{R}^n$).

For example, we can use co-occurrence within ax^\vdash to constrain the embedding function, where the constraints on co-occurrence are formulated using the Word2Vec skipgram model.

Onto2Vec

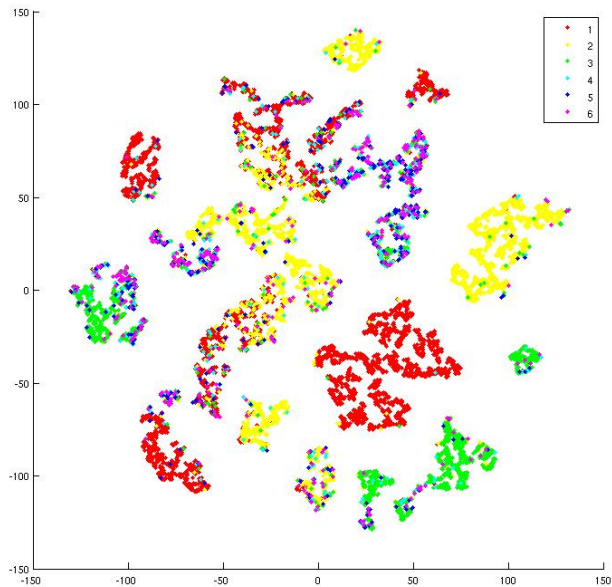


Predicting PPIs: trainable similarity measures



Smaili et al. Onto2Vec: joint vector-based representation of biological entities and their ontology-based annotations.

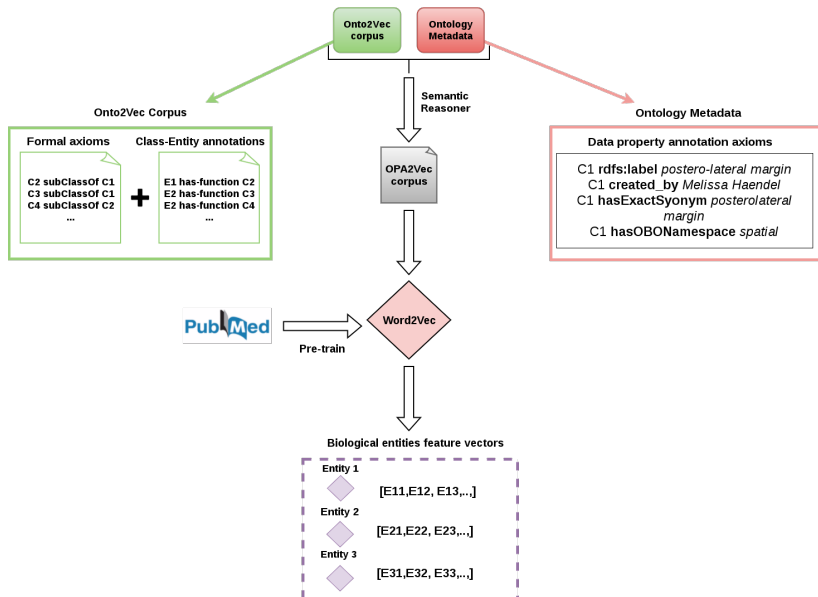
Visualizing embeddings



Combination with text

- ontologies contain more than axioms:
 - ▶ labels, synonyms, definitions, authors, etc.
- Description Logic axioms \neq natural language
- transfer learning: learn on one domain/task, apply to another
 - ▶ e.g.: learn on literature, apply to ontologies
 - ▶ words have “meaning” in literature, Description Logic symbols have “meaning” in ontology axioms
- Ontologies Plus Annotations 2 Vec (OPA2Vec) combines both

Ontologies Plus Annotations 2 Vec



Axioms contribute to prediction tasks: GO and GO-PLUS

	Human	Yeast	Arabidopsis
<i>GO_Onto2Vec</i>	0.7660	0.7701	0.7559
<i>GO_Onto2Vec_NN</i>	0.8779	0.8711	0.8364
<i>GO_plus_Onto2Vec</i>	0.7880	0.7943	0.7889
<i>GO_plus_Onto2Vec_NN</i>	0.9021	0.8937	0.8834

Evaluating individual axioms

Testing how much each ontologies' axioms contribute to predictions:

	Human		Arabidopsis	
	Onto2Vec	Onto2Vec_NN	Onto2Vec	Onto2Vec_NN
GO (Baseline)	0.7660	0.8779	0.7559	0.8364
ChEBI	0.7899(+0.0239)	0.8914(+0.0135)	0.7703(+0.0144)	0.8518(+0.0154)
PO	0.7752(+0.0092)	0.8776(-0.0003)	0.7671(+0.0112)	0.8469(+0.0105)
CL	0.7743(+0.0083)	0.8810(+0.0031)	0.7612(+0.0053)	0.8371(+0.0007)
PATO	0.7657(-0.0003)	0.8768(-0.0011)	0.7563(+0.0004)	0.8380(+0.0016)

Evaluating definitions

Testing how much each ontologies' annotation properties contribute to predictions:

	Human		Arabidopsis	
	Onto2Vec	Onto2Vec_NN	Onto2Vec	Onto2Vec_NN
GO (Baseline)	0.8727	0.9033	0.8613	0.8903
ChEBI	0.8571(-0.0156)	0.8801(-0.0232)	0.8601(-0.0012)	0.8880(-0.0023)
PO	0.8680(-0.0047)	0.8824(-0.0209)	0.8632(+0.0019)	0.8908(+0.0005)
CL	0.8811(+0.0084)	0.9037(+0.0004)	0.8614(+0.0001)	0.8899(-0.0004)
PATO	0.8562(-0.0165)	0.8711(-0.0322)	0.8544(-0.0069)	0.8860(-0.0043)

- `https://github.com/bio-ontology-research-group/opa2vec`
- command line tool
 - ▶ input: OWL ontology, set of entities with annotations/associations
 - ▶ output: vectors for each class and entity
- includes Elk and HermiT
- limitations: word-based
 - ▶ still ignores semantics!

OPA2Vec Jupyter exercise

- open the notebook `OPA2Vec.ipynb`
- run the whole notebook
 - ▶ this should be relatively fast and not take too much time on a modern laptop
- play with the prediction methods (cosine similarity)

How to overcome the semantic gap?

- none of the models discussed above are truly “semantic”
 - ▶ all syntactic
 - ▶ graph-based or based on axioms

How to overcome the semantic gap?

- none of the models discussed above are truly “semantic”
 - ▶ all syntactic
 - ▶ graph-based or based on axioms
- what do we actually mean by “semantics”?

How to overcome the semantic gap?

- none of the models discussed above are truly “semantic”
 - ▶ all syntactic
 - ▶ graph-based or based on axioms
- what do we actually mean by “semantics”?
 - ▶ formal definition of “truth” relies on “models”

How to overcome the semantic gap?

- none of the models discussed above are truly “semantic”
 - ▶ all syntactic
 - ▶ graph-based or based on axioms
- what do we actually mean by “semantics”?
 - ▶ formal definition of “truth” relies on “models”
 - ▶ universal algebra over formal languages (with signature Σ)

Description Logic EL++

Name	Syntax	Semantics
top	\top	$\Delta^{\mathcal{I}}$
bottom	\perp	\emptyset
nominal	$\{a\}$	$\{a^{\mathcal{I}}\}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
generalized concept inclusion	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
role inclusion	$r_1 \circ \dots \circ r_n \sqsubseteq r$	$r_1^{\mathcal{I}} \circ \dots \circ r_n^{\mathcal{I}} \subseteq r^{\mathcal{I}}$

- Interpretations and Σ -structures
- Model \mathfrak{A} of a formula ϕ : ϕ is true in \mathfrak{A} ($\mathfrak{A} \models \phi$)
- Theory T : set of formulas
- \mathfrak{A} is a model of T if \mathfrak{A} is a model of all formulas in T
- Ontologies are (special kinds of) theories

EL Embeddings

- given a theory/ontology T with signature $\Sigma(T)$
- aim: find $f_e : \Sigma(T) \mapsto \mathbb{R}^n$ s.t. $f_e(\Sigma(T))$ is a model of T
($f_e(\Sigma(T)) \models T$)

EL Embeddings

- given a theory/ontology T with signature $\Sigma(T)$
- aim: find $f_e : \Sigma(T) \mapsto \mathbb{R}^n$ s.t. $f_e(\Sigma(T))$ is a model of T
($f_e(\Sigma(T)) \models T$)
- more general: find an algorithm that maps symbols (signatures) into \mathbb{R}^n so that the *semantics* of the symbol (expressed through axioms and explicit in model structures) is preserved

EL Embeddings

- given a theory/ontology T with signature $\Sigma(T)$
- aim: find $f_e : \Sigma(T) \mapsto \mathbb{R}^n$ s.t. $f_e(\Sigma(T))$ is a model of T
($f_e(\Sigma(T)) \models T$)
- more general: find an algorithm that maps symbols (signatures) into \mathbb{R}^n so that the *semantics* of the symbol (expressed through axioms and explicit in model structures) is preserved
- any consistent \mathcal{EL}^{++} theory has infinite models

EL Embeddings

- given a theory/ontology T with signature $\Sigma(T)$
- aim: find $f_e : \Sigma(T) \mapsto \mathbb{R}^n$ s.t. $f_e(\Sigma(T))$ is a model of T
($f_e(\Sigma(T)) \models T$)
- more general: find an algorithm that maps symbols (signatures) into \mathbb{R}^n so that the *semantics* of the symbol (expressed through axioms and explicit in model structures) is preserved
- any consistent \mathcal{EL}^{++} theory has infinite models
- any consistent EL^{++} theory has models in \mathbb{R}^n (Loewenheim-Skolem, upwards)

Key idea

- for all $r \in \Sigma(T)$ and $C \in \Sigma(T)$, define $f_e(r)$ and $f_e(C)$
- $f_e(C)$ maps to points in an open n -ball such that $f_e(C) = C^{\mathcal{I}}$:
 $C^{\mathcal{I}} = \{x \in \mathbb{R}^n \mid \|f_e(C) - x\| < r_e(C)\}$
 - ▶ these are the *extension* of a class in \mathbb{R}^n
- $f_e(r)$ maps a binary relation r to a vector such that
 $r^{\mathcal{I}} = \{(x, y) \mid x + f_e(r) = y\}$
 - ▶ that's the TransE property for *individuals*
- use the axioms in T as constraints

Algorithm

- normalize the theory:
 - ▶ every \mathcal{EL}^{++} theory can be expressed using four normal forms (Baader et al., 2005)
- eliminate the ABox: replace each individual symbol with a singleton class: a becomes $\{a\}$
- rewrite relation assertions $r(a, b)$ and class assertions $C(a)$ as $\{a\} \sqsubseteq \exists r. \{b\}$ and $\{a\} \sqsubseteq C$
 - ▶ something to remember for the next class-vs-instance discussion?
- normalization rules to generate:
 - ▶ $C \sqsubseteq D$
 - ▶ $C \sqcap D \sqsubseteq E$
 - ▶ $C \sqsubseteq \exists R.D$
 - ▶ $\exists R.C \sqsubseteq D$

Algorithm: loss functions

$$\begin{aligned} \text{loss}_{C \sqsubseteq D}(c, d) = & \\ & \max(0, \|f_\eta(c) - f_\eta(d)\| + r_\eta(c) - r_\eta(d) - \gamma) \\ & + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \end{aligned} \quad (3)$$

Algorithm: loss functions

Let $h = \frac{r_\eta(c)^2 - r_\eta(d)^2 + \|f_\eta(c) - f_\eta(d)\|^2}{2\|f_\eta(c) - f_\eta(d)\|}$, then the center and radius of the smallest n -ball containing the intersection of $\eta(C)$ and $\eta(D)$ are $f_\eta(c) + \frac{h}{\|f_\eta(c) - f_\eta(d)\|}(f_\eta(d) - f_\eta(c))$ and $\sqrt{r_\eta(c)^2 - h^2}$.

Algorithm: loss functions

$$\begin{aligned} \text{loss}_{C \sqsubseteq \exists R.D}(c, d, r) = \\ \max(0, \|f_\eta(c) + f_\eta(r) - f_\eta(d)\| + r_\eta(c) - r_\eta(d) - \gamma) \quad (4) \\ + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \end{aligned}$$

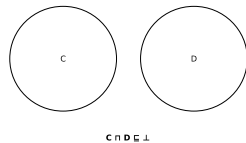
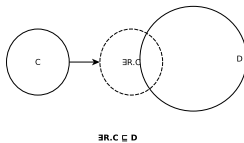
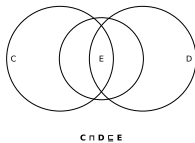
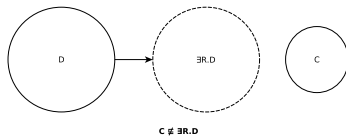
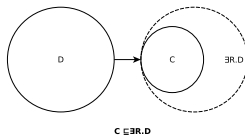
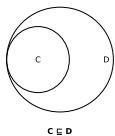
Algorithm: loss functions

$$\begin{aligned} \text{loss}_{\exists R.C \sqsubseteq D}(c, d, r) = \\ \max(0, \|f_\eta(c) - f_\eta(r) - f_\eta(d)\| - r_\eta(c) - r_\eta(d) - \gamma) \quad (5) \\ + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \end{aligned}$$

Algorithm: loss functions

$$\begin{aligned} \text{loss}_{C \cap D \sqsubseteq \perp}(c, d, e) = \\ \max(0, r_\eta(c) + r_\eta(d) - \|f_\eta(c) - f_\eta(d)\| + \gamma) \\ + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \end{aligned} \quad (6)$$

Algorithm: loss functions



EL Embeddings

Male \sqsubseteq *Person* (7)

Female \sqsubseteq *Person* (8)

Father \sqsubseteq *Male* (9)

Mother \sqsubseteq *Female* (10)

Father \sqsubseteq *Parent* (11)

Mother \sqsubseteq *Parent* (12)

Female \sqcap *Male* $\sqsubseteq \perp$ (13)

Female \sqcap *Parent* \sqsubseteq *Mother* (14)

Male \sqcap *Parent* \sqsubseteq *Father* (15)

$\exists hasChild. Person$ \sqsubseteq *Parent* (16)

Parent \sqsubseteq *Person* (17)

Parent $\sqsubseteq \exists hasChild. \top$ (18)

EL Embeddings

- model with $\Delta = R^n$
- support quantifiers, negation, conjunction,...

Jupyter exercise

- Run the new Docker image
coolmaksat/embeddings:latest
- `docker run -i -t -p 8888:8888
coolmaksat/embeddings /bin/bash -c "jupyter
notebook --notebook-dir=/usr/src/app/
--ip='0.0.0.0' --port=8888 --no-browser
--allow-root"`

Summary

- ontologies contain background knowledge that is useful as background knowledge:
 - ▶ axioms
 - ▶ natural language (definitions, labels, synonyms)

Summary

- ontologies contain background knowledge that is useful as background knowledge:
 - ▶ axioms
 - ▶ natural language (definitions, labels, synonyms)
- feature learning (deep learning) on ontologies encodes this background knowledge
 - ▶ using ontology graphs, axioms, or model structures

Open research questions

Where is our semantics, in the machine learning model or the axioms?

- implicit or explicit?
- hidden or interpretable?
- example: transitive relations
- combination of both?

Open research questions

How do we evaluate our models and methods?

- random splits (standard in machine learning)
- time-based splits
- other?

Open research questions

What is the interface between knowledge representation and learning?

- *How* to represent knowledge affects learning outcomes
- representation patterns and learning \Rightarrow specific algorithms?

Acknowledgements

Lots of help from my students:

- Mona Alshahrani
- Maxat Kulmanov
- Fatima Smaili

and colleagues:

- Mehdi Ali (PyKEEN)
- Michel Dumontier
- Andreas Karwath
- Paul Schofield

and KAUST (Bio2Vec grant, award no URF/1/3454-01-01).