Semantic similarity and machine learning with ontologies

Robert Hoehndorf and Maxat Kulmanov

Embedding ontologies: approaches

- syntactic: treat axioms as “sentences” using language models
- graph-based: treat ontologies as graphs (like in semantic similarity)
- model-theoretic: encode model-theoretic semantics in optimization

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 $\Sigma(O) \mapsto \mathbb{R}^n$ (subject to certain constraints).

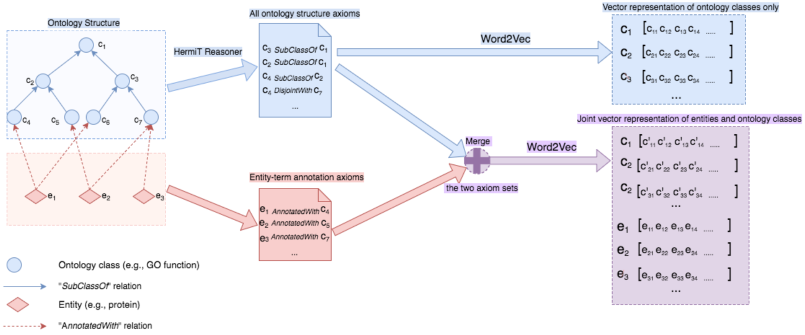
Ontology embeddings

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 $\Sigma(O) \mapsto \mathbb{R}^n$ (subject to certain constraints).

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



Ontology Structure
HermiT Reasoner
All ontology structure axioms
c3 SubClassOf c1
c2 SubClassOf c1
c4 SubClassOf c2
c4 DisjointWith c7
Word2Vec
Vector representation of ontology classes only
Entity-term annotation axioms
e1 AnnotatedWith c4
e2 AnnotatedWith c5
e3 AnnotatedWith c7
Merge
the two axiom sets
Word2Vec
Joint vector representation of entities and ontology classes
Ontology class (e.g., GO function)
"SubClassOf" relation
Entity (e.g., protein)
"AnnotatedWith" relation

Word2Vec

Maximize:

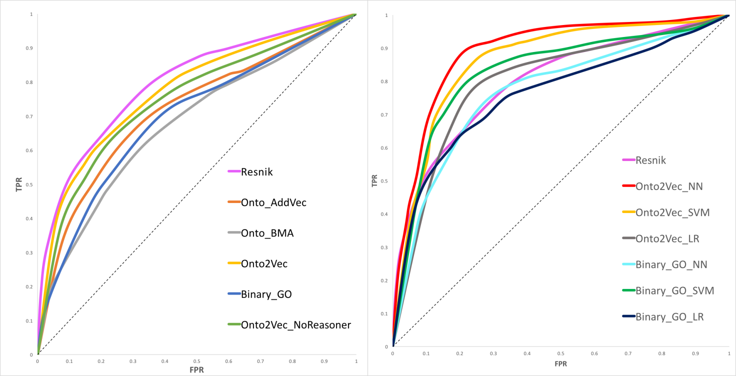
$$\frac{1}{N}\sum_{n=1}^{N}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{n+j}|w_n) \tag{1}$$

with

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

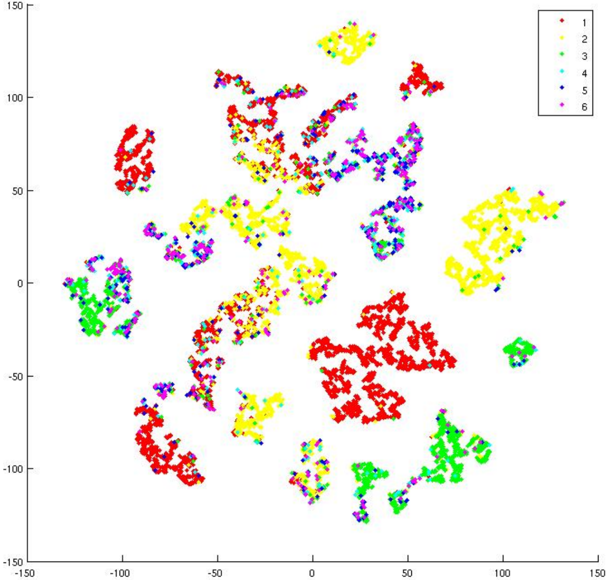
(at least conceptually; different strategies are used to approximate Eqn. 2)

Predicting PPIs: trainable similarity measures



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

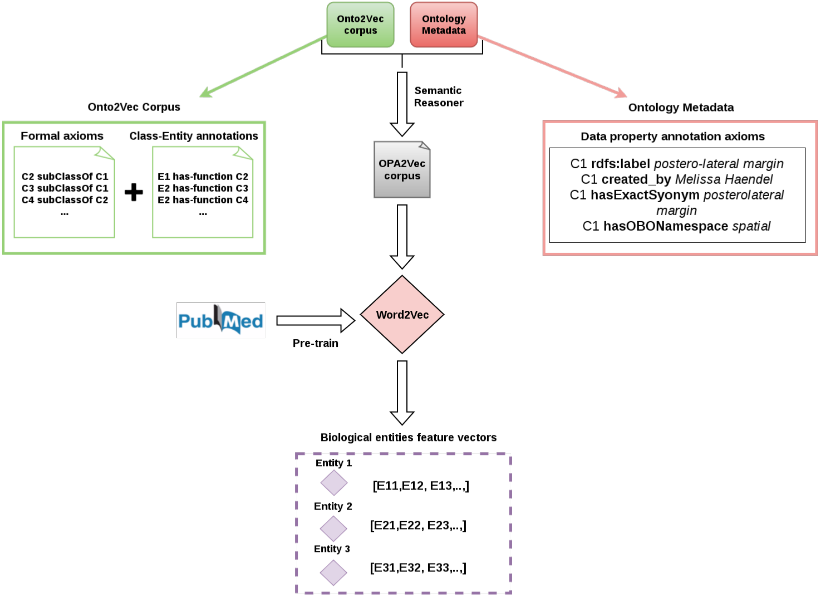
Visualizing embeddings



Combination with text

- ontologies contain more than axioms:
  - labels, synonyms, definitions, authors, etc.
- Description Logic axioms != 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
GO_Onto2Vec	0.7660	0.7701	0.7559
GO_Onto2Vec_NN	0.8779	0.8711	0.8364
GO_plus_Onto2Vec	0.7880	0.7943	0.7889
GO_plus_Onto2Vec_NN	**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)

OPA2Vec

- 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
  - completely ignores any semantics!

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 $f_\eta : L(V) \cup L(E) \mapsto \mathrm{R}^n$.

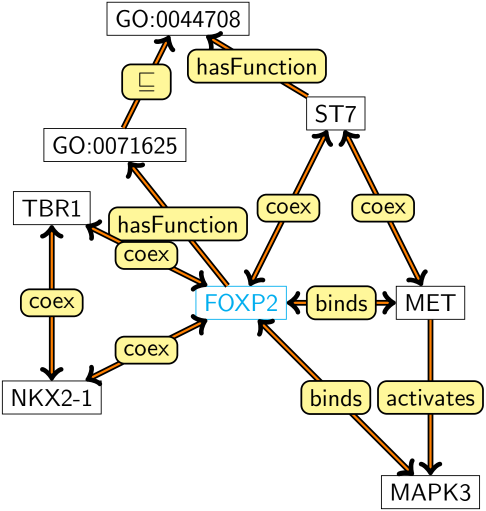
Graph embeddings

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 $f_\eta : L(V) \cup L(E) \mapsto \mathrm{R}^n$.

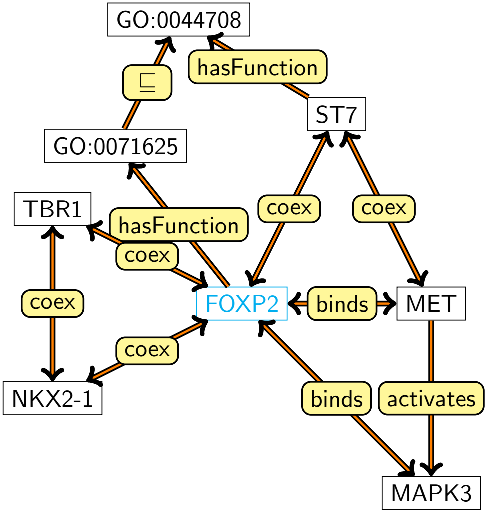
- key idea: preserve *some* structure of the graph in $\mathbb{R}^n$ (under operations in $\mathbb{R}^n$)
- $\mathbb{R}^n$ enables *new* operations (such as many similarity measures)
- useful as *feature* vectors

Random walks



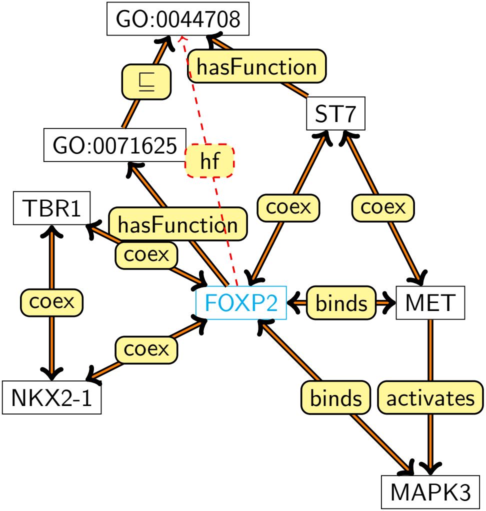
- FOXP2 is characterized by *adjacent* and close nodes and edges
- different edges may "transmit" information differently

Random walks



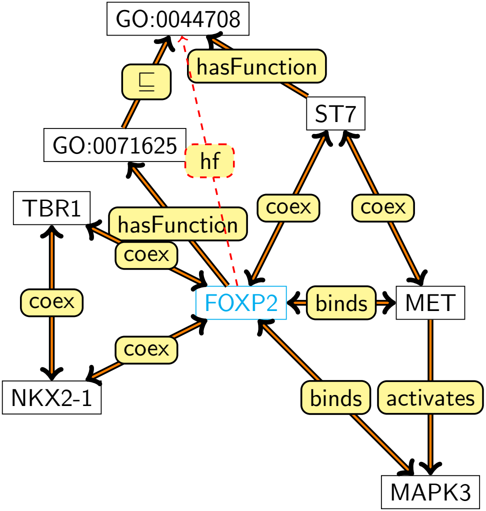
- precompute the deductive closure:
- for all ϕ: if $\mathcal{KG} \models \phi$, add ϕ to $\mathcal{KG}$

Random walks



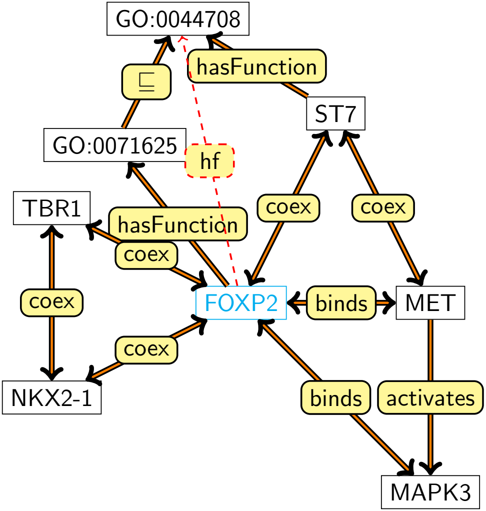
- precompute the deductive closure:
- for all ϕ: if $\mathcal{KG} \models \phi$, add ϕ to $\mathcal{KG}$

Random walks



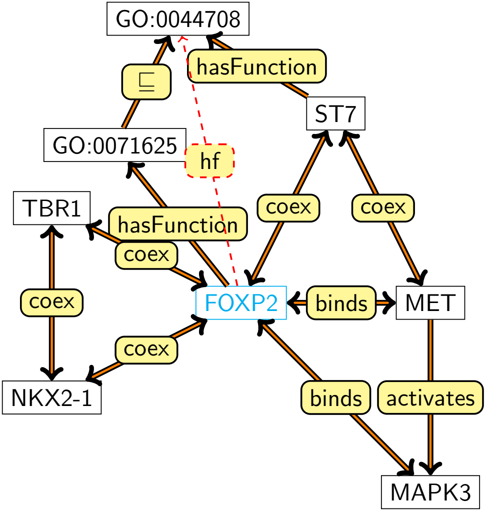
- Exploring the graph:

Random walks



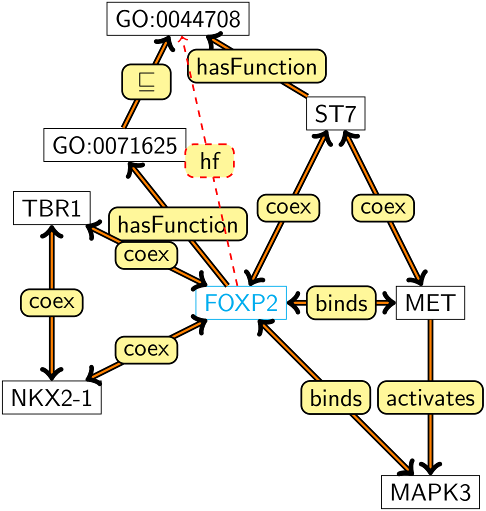
- Exploring the graph:
- :FOXP2 :binds :MET :coex :ST7 :hasFunction GO:0044708

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 ...

Word2Vec and Random Walks

- random walks "flatten" a graph
  - walks capture node neighborhood
  - and generate a "corpus"
- random walks capture graph "structure"
  - in ABox and TBox
  - 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

Ontologies, graphs, and text

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.

Ontologies, graphs, and text

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 ...

Ontologies, graphs, and text

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 ...

Tools and resources

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

Tools and resources

- 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"

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 $f_\eta : L(V) \cup L(E) \mapsto \mathrm{R}^n$.

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 $f_\eta : L(V) \cup L(E) \mapsto \mathrm{R}^n$.

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 $f_\eta : L(V) \cup L(E) \mapsto \mathrm{R}^n$.

Graph as edgelist: set of (s, p, o) statements
Idea: $\mu(s) + \mu(p) \approx \mu(o)$

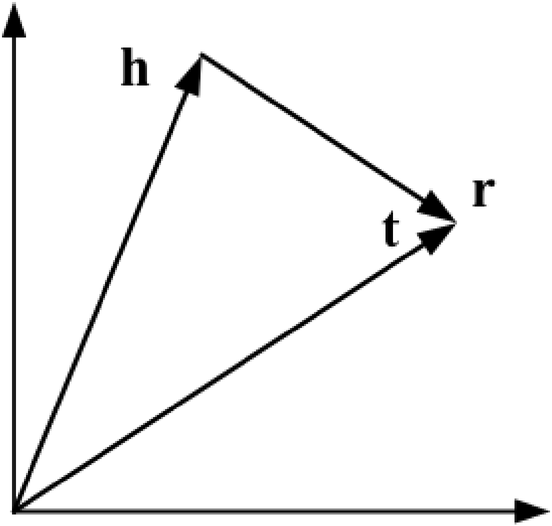
Translating embeddings

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 $f_\eta : L(V) \cup L(E) \mapsto \mathrm{R}^n$.

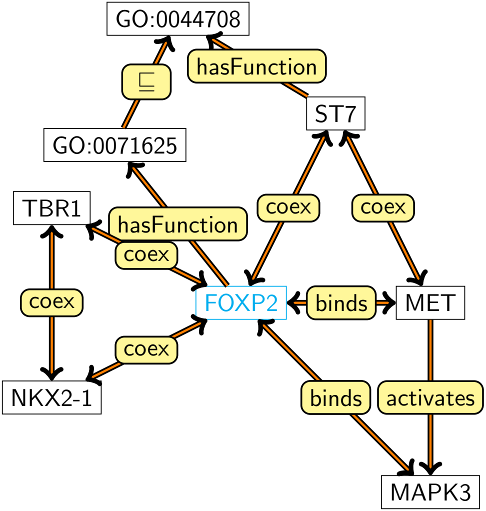
Graph as edgelist: set of (s, p, o) statements
Idea: $\mu(s) + \mu(p) \approx \mu(o)$
Minimize: $\sum_t \|\mu(s) + \mu(p) - \mu(o)\|$ (chose your norm, usually L2)

Translating embeddings

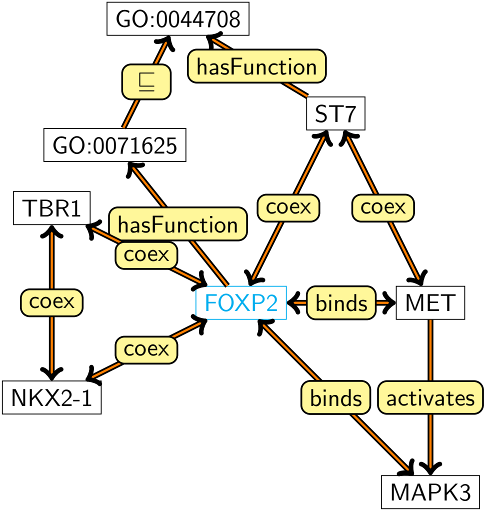


Entity and Relation Space

Translating embeddings

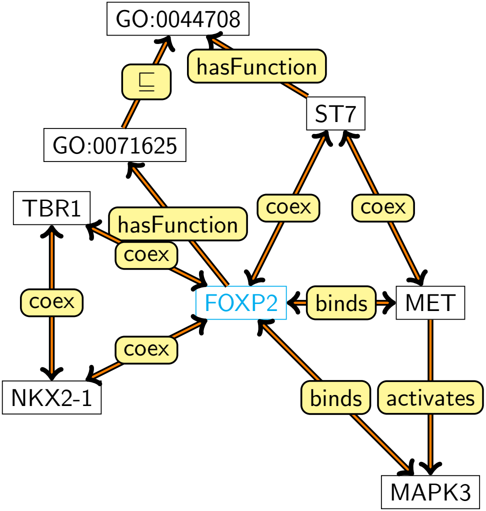


Translating embeddings



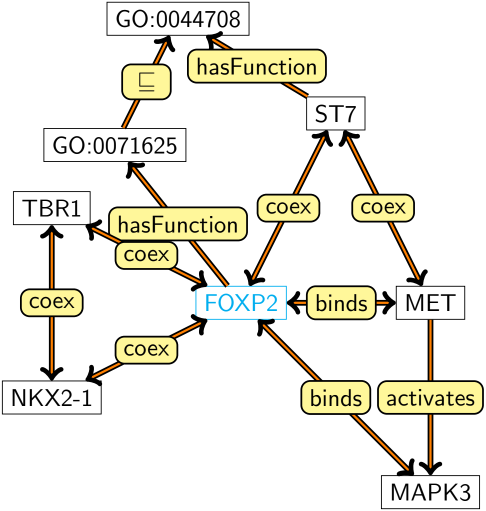
- FOXP2 + binds = MET

Translating embeddings



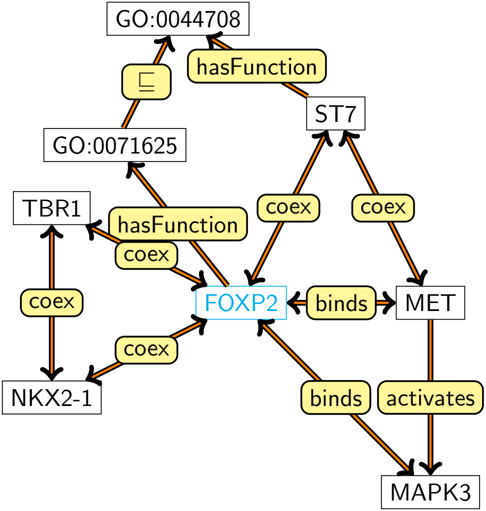
- FOXP2 + binds = MET
- MET + activates = MAPK3

Translating embeddings



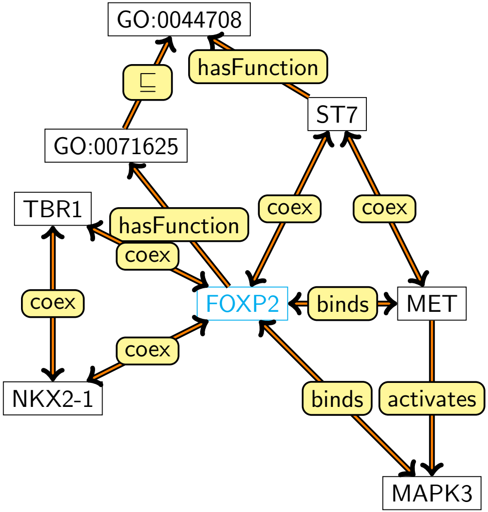
- FOXP2 + binds = MET
- MET + activates = MAPK3
- MET + binds = FOXP2

Translating embeddings



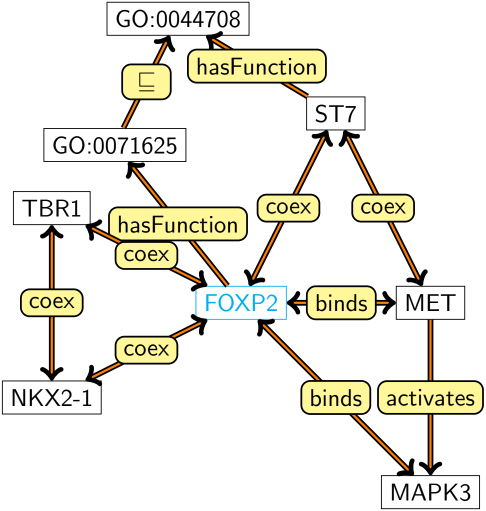
- FOXP2 + binds = MET
- MET + activates = MAPK3
- MET + binds = FOXP2
- ST7 + hasFunction = `GO:0044708`

Translating embeddings



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

Translating embeddings

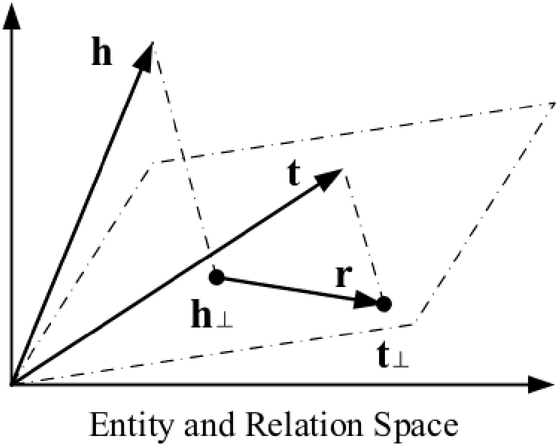


- FOXP2 + binds - MET = 0
- MAP + activates - MAPK3 = 0
- MET + binds - FOXP2 = 0
- ST7 + hasFunction - `GO:0044708` = 0
- ...

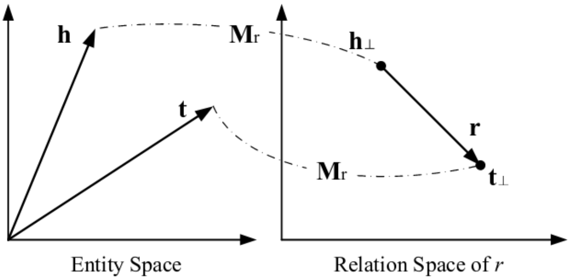
Some properties of TransE

- 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.

PyKEEN

- 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
- (this also holds for Graph Convolutional Networks, which are not covered here)

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	$\bot$	$\emptyset$
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	$r_1 \circ \ldots \circ r_n \sqsubseteq r$	$r_1^{\mathcal{I}} \circ \ldots \circ r_n^{\mathcal{I}} \subseteq r^{\mathcal{I}}$

Models

- 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
  - or: the embedding function *is* an interpretation function

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
  - or: the embedding function *is* an interpretation function
- 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
  - or: the embedding function *is* an interpretation function
- any consistent $\mathcal{EL}^{++}$theory has infinite models
- any consistent $\mathcal{EL}^{++}$theory has models in $\mathbb{R}^n$ (Loewenheim-Skolem, upwards; compactness)

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 $f_e(C) = C^{\mathcal{I}}$: $C^{\mathcal{I}} = \{x \in \mathbb{R}^n | \, \|f_e(C) - x\| < r_e(C)\}$
  - these are the *extension* of a class in $\mathbb{R}^n$
- $f_e(r)$ maps a binary relation r to a vector such that $r^{\mathcal{I}} = \{(x, y) | x + f_e(r) = y\}$
  - 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

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

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

$$loss_{C \sqsubseteq \exists R.D}(c, d, r) =$$
$$\max(0, \|f_\eta(c) + f_\eta(r) - f_\eta(d)\| + r_\eta(c) - r_\eta(d) - \gamma) \tag{4}$$
$$+ |\,\|f_\eta(c)\| - 1| + |\,\|f_\eta(d)\| - 1|$$

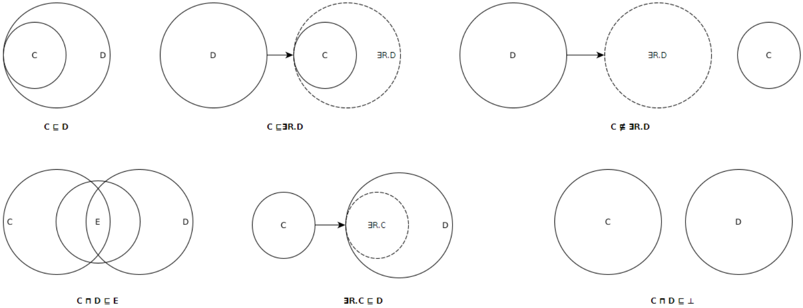
Algorithm: loss functions

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

Algorithm: loss functions

$$loss_{C \sqcap D \sqsubseteq \bot}(c, d, e) =$$
$$\max(0, r_\eta(c) + r_\eta(d) - \|f_\eta(c) - f_\eta(d)\| + \gamma) \tag{6}$$
$$+ \left| \|f_\eta(c)\| - 1 \right| + \left| \|f_\eta(d)\| - 1 \right|$$

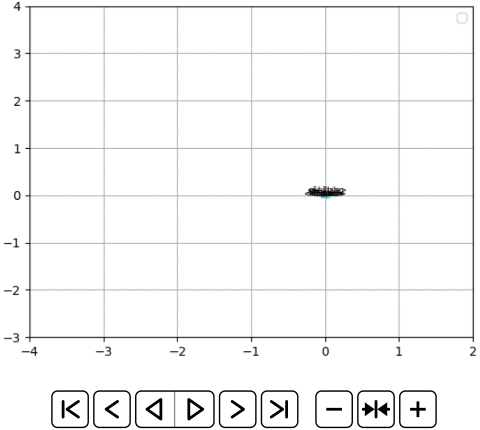
Algorithm: loss functions



EL Embeddings

$$
\begin{aligned}
\mathit{Male} &\sqsubseteq \mathit{Person} && (7)\\
\mathit{Female} &\sqsubseteq \mathit{Person} && (8)\\
\mathit{Father} &\sqsubseteq \mathit{Male} && (9)\\
\mathit{Mother} &\sqsubseteq \mathit{Female} && (10)\\
\mathit{Father} &\sqsubseteq \mathit{Parent} && (11)\\
\mathit{Mother} &\sqsubseteq \mathit{Parent} && (12)\\
\mathit{Female} \sqcap \mathit{Male} &\sqsubseteq \bot && (13)\\
\mathit{Female} \sqcap \mathit{Parent} &\sqsubseteq \mathit{Mother} && (14)\\
\mathit{Male} \sqcap \mathit{Parent} &\sqsubseteq \mathit{Father} && (15)\\
\exists \mathit{hasChild}.\mathit{Person} &\sqsubseteq \mathit{Parent} && (16)\\
\mathit{Parent} &\sqsubseteq \mathit{Person} && (17)\\
\mathit{Parent} &\sqsubseteq \exists \mathit{hasChild}.\top && (18)
\end{aligned}
$$

EL Embeddings



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