

# Semantic similarity and machine learning with ontologies

A brief introduction

Robert Hoehndorf

`https://github.com/bio-ontology-research-group/  
ontology-tutorial`

## 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

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 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
  - ▶ graph-based: ontology as a graph
  - ▶ feature-based: extract (or obtain) features for classes/relations (e.g., from axioms)
  - ▶ model-based: define similarity within (special)  $\Sigma$ -structures

# Ontologies and graphs

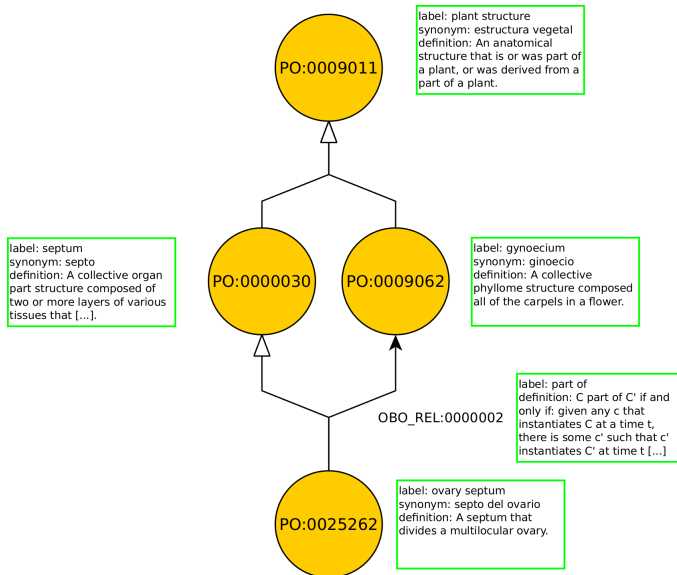
- semantic similarity measures *and machine learning models* on ontologies can be graph-based, feature-based, or model-based
  - ▶ graph-based: ontology as a graph
  - ▶ feature-based: extract (or obtain) features for classes/relations (e.g., from axioms)
  - ▶ model-based: define similarity within (special)  $\Sigma$ -structures
- 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
  - ▶ graph-based: ontology as a graph
  - ▶ feature-based: extract (or obtain) features for classes/relations (e.g., from axioms)
  - ▶ model-based: define similarity within (special)  $\Sigma$ -structures
- 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
  - ▶ design patterns as “relations” between classes



# Relations as patterns



# 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:  $Y: X \xleftrightarrow{=} Y, \{X, Y\}$
- ...

NB: in bio-ontologies, the OBO Relation Ontology defines these patterns

# Asserted and inferred

- relation patterns can be asserted or inferred
- $X \text{ SubClassOf: part-of some } Y$
- $Y \text{ SubClassOf: part-of some } Z$
- $\text{part-of} \circ \text{part-of SubPropertyOf: part-of}$
- $\vdash X \text{ SubClassOf: part-of some } Z$
- Therefore:  $X \xrightarrow{\text{part-of}} Z$
- $\Rightarrow$  we should use deductive inference to generate these patterns

## Semantic similarity

- We want to use *background knowledge* in ontologies to
  - ▶ determine similarity between classes,
  - ▶ instances,
  - ▶ and entities with ontology annotations

# How to measure similarity?

- semantic similarity measures similarity between classes
- semantic similarity measures similarity between instances of classes
- semantic similarity measures similarity between entities *annotated* with classes
- $\Rightarrow$  reduce all of this to similarity between classes

# How to measure similarity?

What properties do we want in a similarity measure?

A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

# How to measure similarity?

What properties do we want in a similarity measure?

A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

- non-negative:  $sim(x, y) \geq 0$  for all  $x, y$

# How to measure similarity?

What properties do we want in a similarity measure?

A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

- non-negative:  $sim(x, y) \geq 0$  for all  $x, y$
- symmetric:  $sim(x, y) = sim(y, x)$



# How to measure similarity?

What properties do we want in a similarity measure?

A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

- non-negative:  $sim(x, y) \geq 0$  for all  $x, y$
- symmetric:  $sim(x, y) = sim(y, x)$
- reflexive:  $sim(x, x) = \max_D$

# How to measure similarity?

What properties do we want in a similarity measure?

A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

- non-negative:  $sim(x, y) \geq 0$  for all  $x, y$
- symmetric:  $sim(x, y) = sim(y, x)$
- reflexive:  $sim(x, x) = \max_D$ 
  - ▶ weaker form:  $sim(x, x) > sim(x, y)$  for all  $x \neq y$

# How to measure similarity?

What properties do we want in a similarity measure?

A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

- non-negative:  $sim(x, y) \geq 0$  for all  $x, y$
- symmetric:  $sim(x, y) = sim(y, x)$
- reflexive:  $sim(x, x) = \max_D$ 
  - ▶ weaker form:  $sim(x, x) > sim(x, y)$  for all  $x \neq y$
- $sim(x, x) > sim(x, y)$  for  $x \neq y$

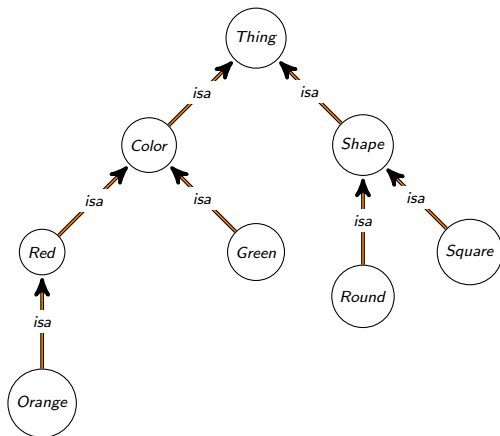
# How to measure similarity?

What properties do we want in a similarity measure?

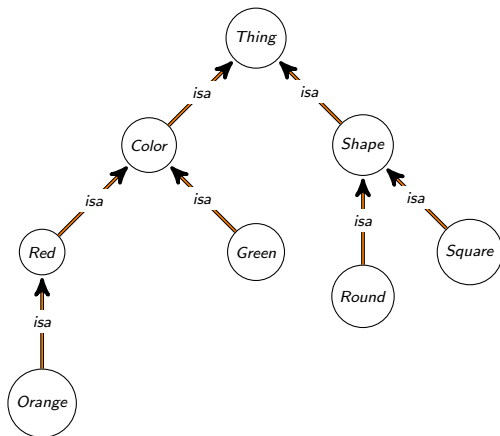
A function  $sim : D \times D$  is a similarity on  $D$  if, for all  $x, y \in D$ , the function  $sim$  is:

- non-negative:  $sim(x, y) \geq 0$  for all  $x, y$
- symmetric:  $sim(x, y) = sim(y, x)$
- reflexive:  $sim(x, x) = \max_D$ 
  - ▶ weaker form:  $sim(x, x) > sim(x, y)$  for all  $x \neq y$
- $sim(x, x) > sim(x, y)$  for  $x \neq y$
- $sim$  is a *normalized* similarity measure if it has values in  $[0, 1]$

# How to measure similarity?

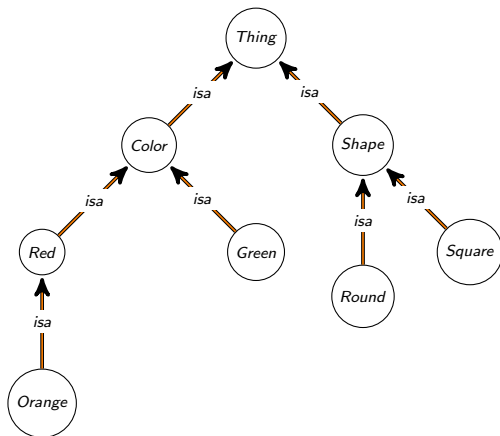


# How to measure similarity?



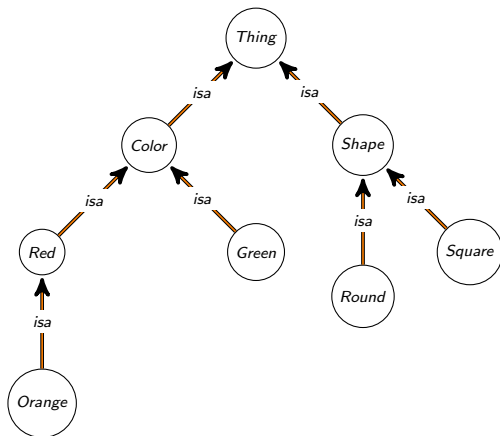
- distance on shortest path (Rada *et al.*, 1989)

# How to measure similarity?



- distance on shortest path (Rada *et al.*, 1989)
- $dist_{Rada}(u, v) = sp(u, isa, v)$

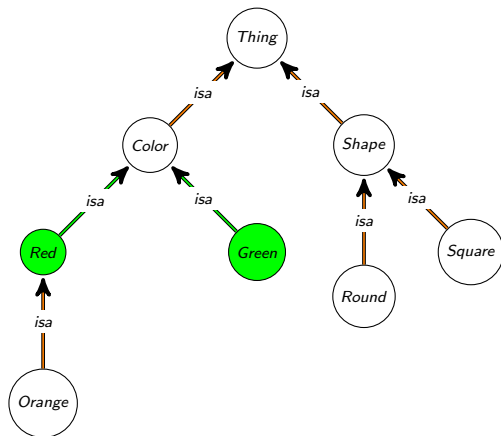
# How to measure similarity?



- distance on shortest path (Rada *et al.*, 1989)
- $dist_{Rada}(u, v) = sp(u, isa, v)$
- $sim_{Rada}(u, v) = \frac{1}{dist_{Rada}(u, v) + 1}$

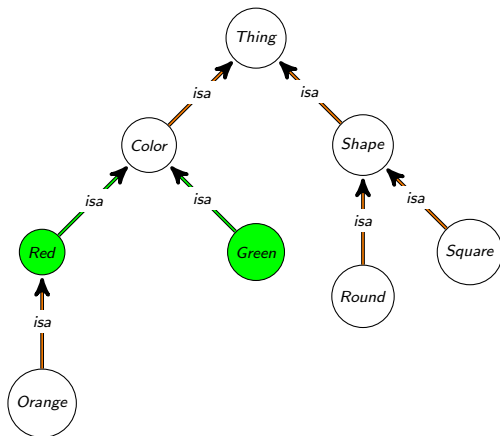


# How to measure similarity?



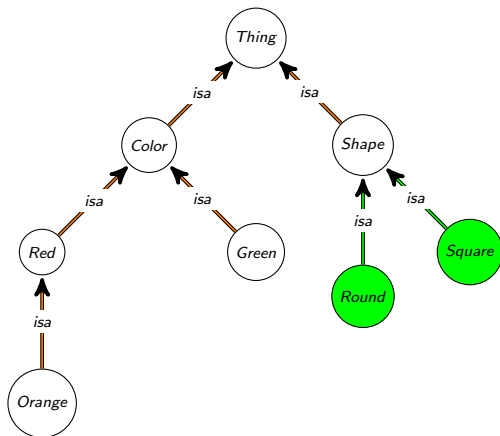
- distance on shortest path

# How to measure similarity?



- distance on shortest path
- $\text{distance}(\text{green}, \text{red}) = 2$
- $\text{sim}_{\text{Rada}}(\text{green}, \text{red}) = \frac{1}{3}$

# How to measure similarity?



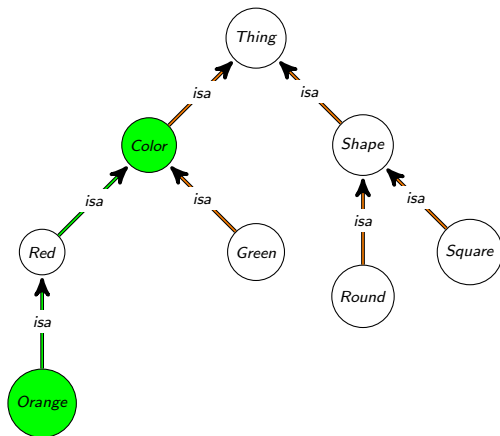
- distance on shortest path

- $\text{distance}(\text{square}, \text{round}) = 2$

- 

$$\text{sim}_{\text{Rada}}(\text{square}, \text{round}) = \frac{1}{3}$$

# How to measure similarity?



- distance on shortest path

- $\text{distance}(\text{orange}, \text{color}) = 2$

- 

$$\text{sim}_{\text{Rada}}(\text{orange}, \text{color}) = \frac{1}{3}$$

# How to measure similarity?

- shortest path is not always intuitive

# How to measure similarity?

- shortest path is not always intuitive
- we need a way to determine *specificity* of a class
  - ▶ number of ancestors
  - ▶ number of children
  - ▶ information content

# How to measure similarity?

- shortest path is not always intuitive
- we need a way to determine *specificity* of a class
  - ▶ number of ancestors
  - ▶ number of children
  - ▶ information content
- *density* of a branch in the ontology
  - ▶ number of siblings
  - ▶ information content

# How to measure similarity?

- shortest path is not always intuitive
- we need a way to determine *specificity* of a class
  - ▶ number of ancestors
  - ▶ number of children
  - ▶ information content
- *density* of a branch in the ontology
  - ▶ number of siblings
  - ▶ information content
- account for different edge types
  - ▶ non-uniform edge weighting



# How to measure similarity

- term specificity measure  $\sigma : \mathcal{C} \mapsto \mathbb{R}$ :
  - ▶  $x \sqsubseteq y \rightarrow \sigma(x) \geq \sigma(y)$

# How to measure similarity

- term specificity measure  $\sigma : C \mapsto \mathbb{R}$ :

- ▶  $x \sqsubseteq y \rightarrow \sigma(x) \geq \sigma(y)$

- intrinsic:

- ▶  $\sigma(x) = f(\text{depth}(x))$

- ▶  $\sigma(x) = f(A(x))$  (for ancestors  $A(x)$ )

- ▶  $\sigma(x) = f(D(x))$  (for descendants  $D(x)$ )

- ▶ many more, e.g., Zhou et al.:

$$\sigma(x) = k \cdot \left(1 - \frac{\log |D(x)|}{\log |C|}\right) + (1 - k) \frac{\log \text{depth}(x)}{\log \text{depth}(G_T)}$$

# How to measure similarity

- term specificity measure  $\sigma : C \mapsto \mathbb{R}$ :

- ▶  $x \sqsubseteq y \rightarrow \sigma(x) \geq \sigma(y)$

- intrinsic:

- ▶  $\sigma(x) = f(\text{depth}(x))$

- ▶  $\sigma(x) = f(A(x))$  (for ancestors  $A(x)$ )

- ▶  $\sigma(x) = f(D(x))$  (for descendants  $D(x)$ )

- ▶ many more, e.g., Zhou et al.:

$$\sigma(x) = k \cdot \left(1 - \frac{\log |D(x)|}{\log |C|}\right) + (1 - k) \frac{\log \text{depth}(x)}{\log \text{depth}(G_T)}$$

- extrinsic:

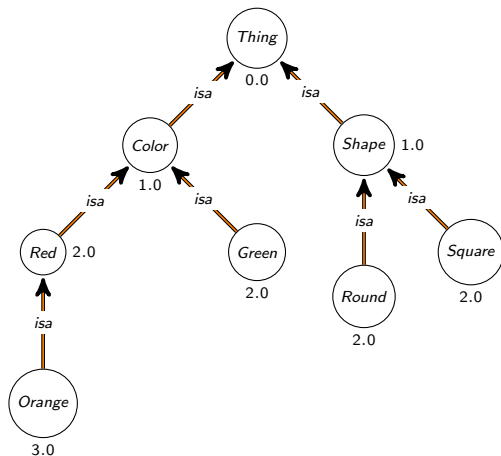
- ▶  $\sigma(x)$  defined as a function of instances (or annotations)  $I$

- ▶ note: the number of instances monotonically decreases with increasing depth in taxonomies

- ▶ Resnik 1995:  $e/C_{\text{Resnik}}(x) = -\log p(x)$  (with  $p(x) = \frac{|I(x)|}{|I|}$ )

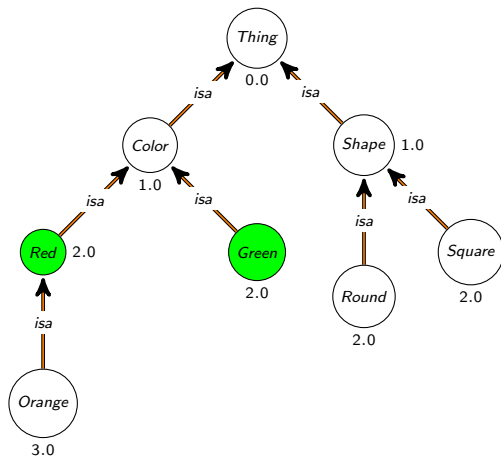
- ▶ in biology, one of the most popular specificity measure when annotations are present

# How to measure similarity?



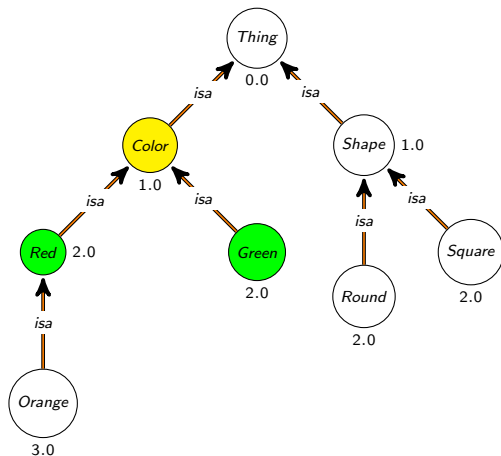
- Resnik 1995:  
similarity between  $x$   
and  $y$  is the  
information content  
of the *most  
informative common  
ancestor*

# How to measure similarity?



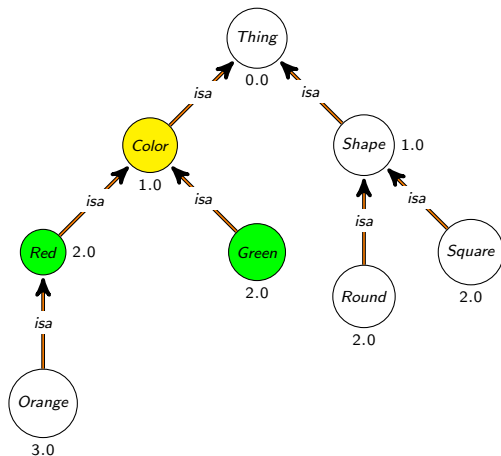
- Resnik 1995:  
similarity between  $x$  and  $y$  is the information content of the *most informative common ancestor*

# How to measure similarity?



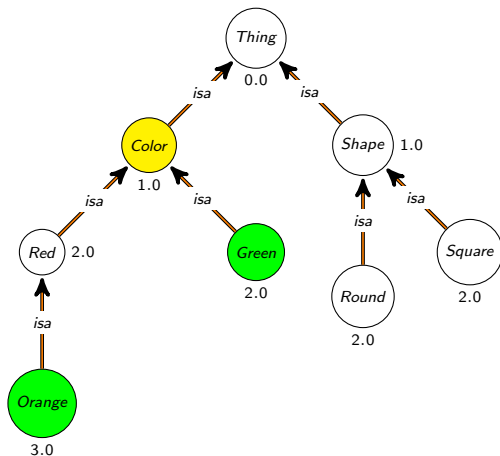
- Resnik 1995:  
similarity between  $x$   
and  $y$  is the  
information content  
of the *most  
informative common  
ancestor*

# How to measure similarity?



- Resnik 1995:  
similarity between  $x$  and  $y$  is the information content of the *most informative common ancestor*
- $$\text{sim}_{\text{Resnik}}(\text{Green}, \text{Red}) = 1.0$$

# How to measure similarity?

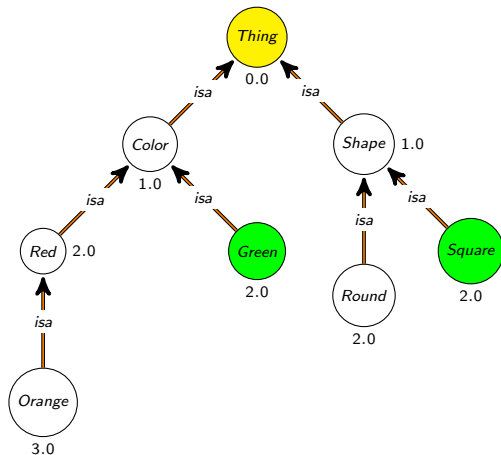


- Resnik 1995:  
similarity between  $x$  and  $y$  is the information content of the *most informative common ancestor*

- $$\text{sim}_{\text{Resnik}}(\text{Green}, \text{Orange}) = 1.0$$



# How to measure similarity?

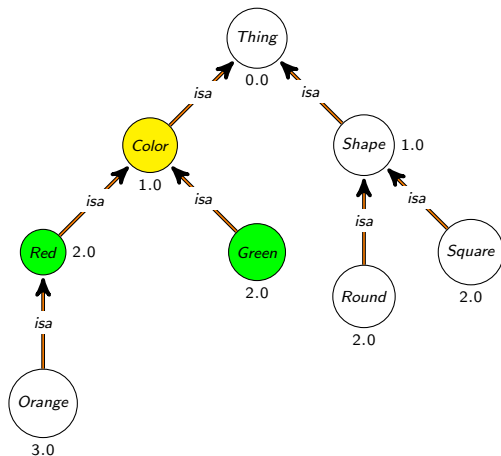


- Resnik 1995:  
similarity between  $x$  and  $y$  is the information content of the *most informative common ancestor*
- $sim_{Resnik}(Square, Orange)$   
0.0

# How to measure similarity?

- (Red, Green) and (Orange, Green) have the same similarity
- need to incorporate the specificity of the compared classes

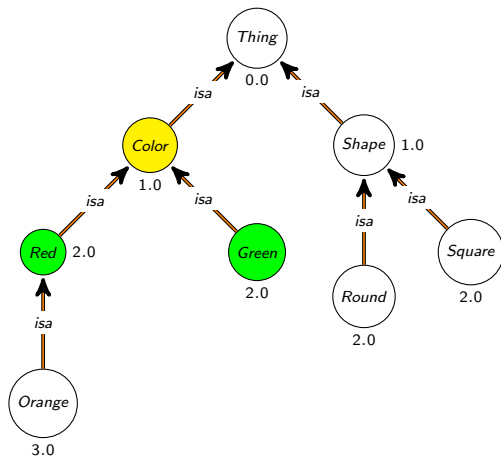
# How to measure similarity?



- Lin 1998:

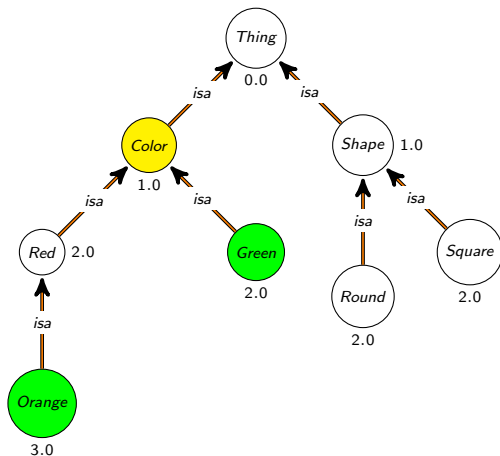
$$sim_{Lin}(x, y) = \frac{2 \cdot IC(MICA(x, y))}{IC(x) + IC(y)}$$

# How to measure similarity?



- Lin 1998:  
$$sim_{Lin}(x, y) = \frac{2 \cdot IC(MICA(x, y))}{IC(x) + IC(y)}$$
- $sim_{Lin}(Green, Red) = 0.5$

# How to measure similarity?



- Lin 1998:

$$sim_{Lin}(x, y) = \frac{2 \cdot IC(MICA(x, y))}{IC(x) + IC(y)}$$

- $sim_{Lin}(Green, Orange) = 0.4$

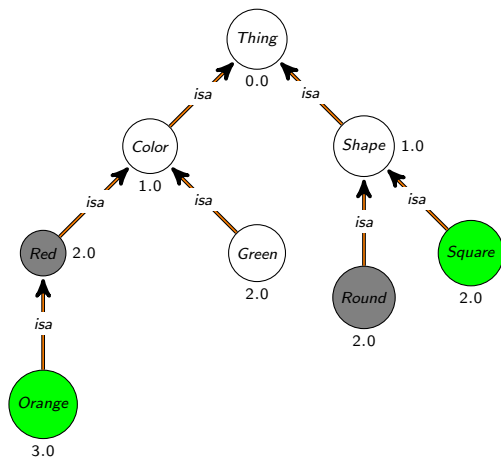
# How to measure similarity?

- many(!) others:
  - ▶ Jiang & Conrath 1997
  - ▶ Mazandu & Mulder 2013
  - ▶ Schlicker et al. 2009
  - ▶ ...

# How to measure similarity?

- we only looked at comparing pairs of classes
- mostly, we want to compare *sets* of classes
  - ▶ set of GO annotations
  - ▶ set of signs and symptoms
  - ▶ set of phenotypes
- two approaches:
  - ▶ compare each class individually, then merge
  - ▶ directly set-based similarity measures

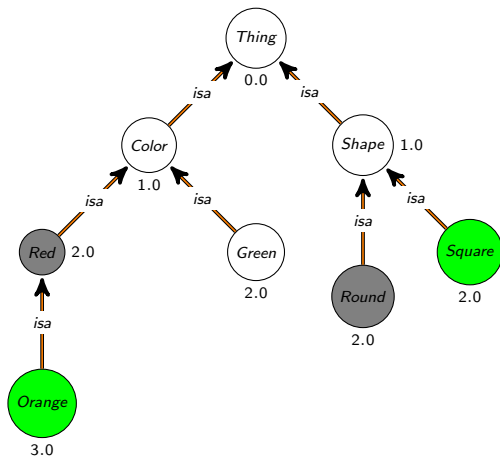
# How to measure similarity?



- similarity between a square-and-orange thing and a round-and-red thing



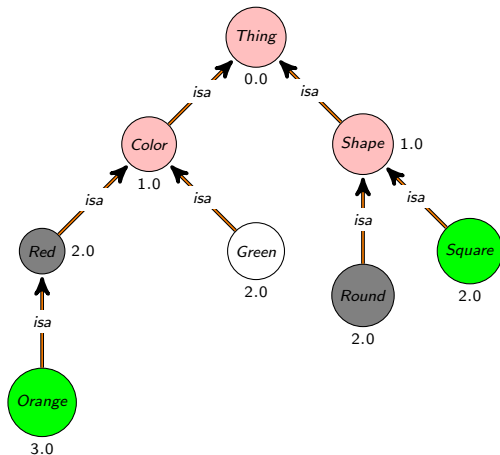
# How to measure similarity?



- similarity between a square-and-orange thing and a round-and-red thing
- Pesquita et al., 2007:

$$\text{simGIC}(X, Y) = \frac{\sum_{c \in A(X) \cap A(Y)} IC(c)}{\sum_{c \in A(X) \cup A(Y)} IC(c)}$$

# How to measure similarity?



- similarity between a square-and-orange thing and a round-and-red thing
- Pesquita et al., 2007:  
$$simGIC(X, Y) = \frac{\sum_{c \in A(X) \cap A(Y)} IC(c)}{\sum_{c \in A(X) \cup A(Y)} IC(c)}$$
- $simGIC(so, rr) = \frac{2}{11}$

# How to measure similarity?

- alternatively: use different merging strategies
- common: average, maximum, **best-matching average**
  - ▶ Average:  $sim_A(X, Y) = \frac{\sum_{x \in X} \sum_{y \in Y} sim(x, y)}{|X| \times |Y|}$
  - ▶ Max average:  $sim_{MA}(X, Y) = \frac{1}{|X|} \sum_{x \in X} \max_{y \in Y} sim(x, y)$
  - ▶ Best match average:  $sim_{BMA}(X, Y) = \frac{sim_{MA}(X, Y) + sim_{MA}(Y, X)}{2}$

# How to measure similarity?

- Semantic Measures Library:
  - ▶ comprehensive Java library
  - ▶ <http://www.semantic-measures-library.org/>
- R packages: GOSim, GOSemSim, HPOSim, LSAfun, ontologySimilarity,...
- Python: sematch, fastsemsim (GO only)

# Applications of semantic similarity

- no obvious choice of similarity measure
- depends on application
  - ▶ e.g., predicting PPIs in different organisms through similarity may benefit from a different similarity measure!
- different similarity measures may react differently to biases in data
- needs some testing and experience

# Applications of semantic similarity

## Recommendations:

- use Resnik's information content measure
- use Resnik's similarity
- use Best Match Average
- use the full ontology
- classify your ontology using an automated reasoner before applying semantic similarity
  - ▶ although many ontologies come pre-classified
- $\Rightarrow$  but there are many exceptions
  - ▶ similar location  $\Rightarrow$  use location subset of GO
  - ▶ developmental phenotypes  $\Rightarrow$  use developmental branch of phenotype ontology

# Analyzing phenotypes

4,000 genetic diseases in OMIM, 6,000 in OrphaNet, have an unknown molecular basis

## OMIM<sup>®</sup>

Online Mendelian Inheritance in Man<sup>®</sup>

An Online Catalog of Human Genes and Genetic Disorders

Number of Entries:					
Prefix	Autosomal	X Linked	Y Linked	Mitochondrial	Totals
* Gene description	12,750	627	48	35	13,460
+ Gene and phenotype, combined	250	14	0	2	266
# Phenotype description, molecular basis known	2,836	240	4	28	3,108
% Phenotype description or locus, molecular basis unknown	1,628	135	5	0	1,768
Other, mainly phenotypes with suspected mendelian basis	1,819	130	2	0	1,951
Totals	19,283	1,146	59	65	20,553

The logo for OrphaNet, featuring the word "orphanet" in a lowercase, sans-serif font. A blue swoosh underline is positioned beneath the letters "anet".




# Analyzing phenotypes

Semantic similarity over phenotype ontologies measures phenotypic similarity

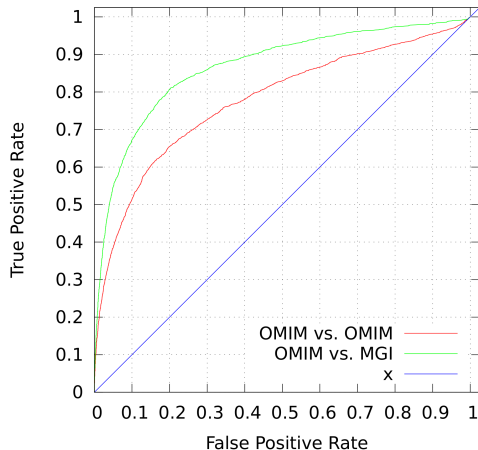
- semantic similarity: similarity measure based on information contained in the axioms/structure of an ontology
  - ▶ anatomy: front limb – hind limb vs. front limb – eye
  - ▶ function: detection of salty taste – detection of sweet taste vs. detection of salty taste – apoptosis
  - ▶ quality: red – orange vs. red – green vs. red – round
- ⇒ phenotypic similarity combines similarity between anatomy, function, and quality



# Analyzing phenotypes

GENOTYPE		
ALSM1(NM_015120.4) [c.10775delC] + [-]	B6.Cg-Alms1 <sup>fox/fox</sup> /J	kcnj11 <sup>c14/c14</sup> ; insr <sup>t143/+</sup> (AB)
		
obesity, diabetes mellitus, insulin resistance	increased food intake, hyperglycemia, insulin resistance	increased weight, adipose tissue volume, glucose homeostasis altered
PHENOTYPE		

# Analyzing phenotypes



- AUC (OMIM): 0.82
- AUC (MGI): 0.90

# Bassoe Syndrome

## Signs and symptoms

- skeletal:
  - ▶ kyphosis, hypertensible joints, cubitus valgus
- muscular:
  - ▶ hypotonia, muscle hypotrophy, amyotrophy
- behavior:
  - ▶ abnormal gait, amimia
- visual:
  - ▶ cataract, strabismus
- reproductive:
  - ▶ hypogonadism, hypogenitalism, abnormal ovaries, hypoplastic testis, reduced fertility

# Bassoe Syndrome

## HIP1 mouse phenotypes

### Bassoe Syndrome:

- kyphosis, hypertensile joints, cubitus valgus
- amyotrophy, hypotonia, muscle hypotrophy
- abnormal gait, amimia
- cataract, strabismus
- testicular atrophy, hypogonadism, hypogenitalism, abnormal ovaries, reduced fertility

### Mouse phenotypes:

- kyphosis, abnormal spine curvature, lordosis
- abnormal muscle morphology
- abnormal gait, hypoactivity, tremors
- nuclear cataracts, microphthalmia
- testicular atrophy, male infertility

An integrative, translational approach to understanding rare and orphan genetically based diseases. Interface Focus, 2013.

# Bassoe Syndrome

## HIP1 mouse phenotypes

### Bassoe Syndrome:

- kyphosis, hypertensible joints, cubitus valgus
- amyotrophy, hypotonia, muscle hypotrophy
- abnormal gait, amimia
- cataract, strabismus
- testicular atrophy, hypogonadism, hypogenitalism, abnormal ovaries, reduced fertility

### Mouse phenotypes:

- kyphosis, abnormal spine curvature, lordosis
- abnormal muscle morphology, **muscle hypotrophy, muscle wasting**
- abnormal gait, hypoactivity, tremors, **failure to thrive, ataxia**
- nuclear cataracts, microphthalmia
- testicular atrophy, male infertility, **ovarian abnormalities, testicular degeneration, increased apoptosis of postmeiotic spermatids, oligospermia**

An integrative, translational approach to understanding rare and orphan genetically based diseases. Interface Focus, 2013.

# Machine learning with 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	<a href="http://purl.obolibrary.org/obo/GO_0001783">http://purl.obolibrary.org/obo/GO_0001783</a>
ontology	GO-PLUS
Equivalent	<a href="#">apoptotic process</a> and ( <a href="#">occurs in some B cell</a> )
SubClassOf	<a href="#">occurs in some B cell</a> , <a href="#">lymphocyte apoptotic process</a>
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  $\Sigma(O) \mapsto \mathbb{R}^n$ ).

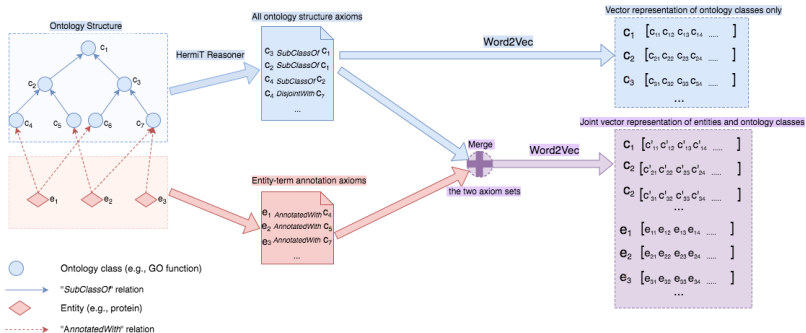
# Ontology embeddings

## 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



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)

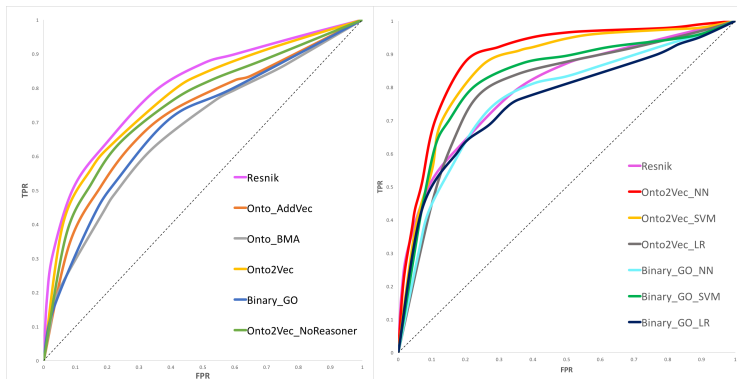
# Word2Vec

## Source Text

## Training Samples

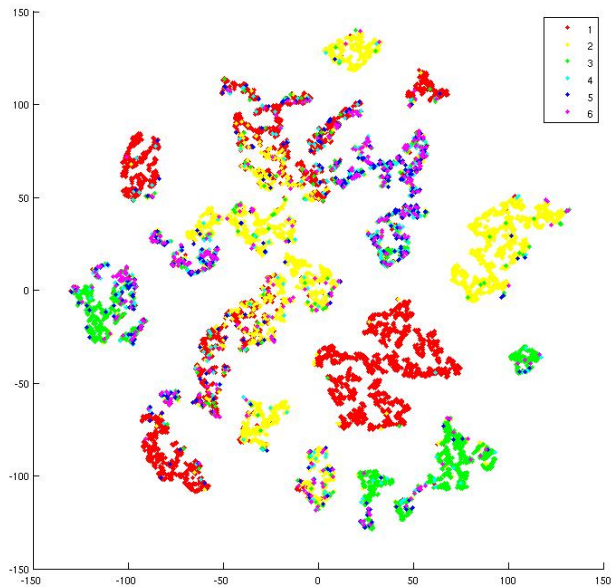
The quick brown fox jumps over the lazy dog. →	(the, quick) (the, brown)
The quick brown fox jumps over the lazy dog. →	(quick, the) (quick, brown) (quick, fox)
The quick brown fox jumps over the lazy dog. →	(brown, the) (brown, quick) (brown, fox) (brown, jumps)
The quick brown fox jumps over the lazy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

# 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>	<b>0.9021</b>	<b>0.8937</b>	<b>0.8834</b>

# 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
  - ▶ completely ignores any semantics!

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

- more slides
  - ▶ ontology theory
  - ▶ machine learning
  - ▶ applications
- code examples, executable notebooks, all dockerized!
  - ▶ predict gene–disease associations through phenotype similarity
  - ▶ protein–protein interactions