Machine learning with ontologies

Preliminaries: ontologies

- An ontology is an explicit [formal] specification of a [shared] conceptualization of a domain.

Preliminaries: ontologies

- An ontology is an explicit [formal] specification of a [shared] conceptualization of a domain.
- what does that mean?

Preliminaries: ontologies

- Ontologies are 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.)
  - artifacts $\rightarrow$ logical theories plus natural language content
  - "understandable" both for humans and for machines
- the intended meaning of *primitive* categories and relations is expressed through axioms (axiomatic method, Tarski)

Preliminaries: ontologies

- Classes and relations: standard identifiers to use across databases (database integration and interoperability)
- Labels: domain vocabulary for classes and relations (natural language processing)
- Metadata and descriptions: definitions/explanations for humans (consistent and competent use of the ontology)
- Axioms and formal definitions: computational access to meaning (automated processing, querying, integration)

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)

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
$\{a\} \sqcup \{b\} \sqcup \ldots$	{a b ...}	{John Robert Mary}

Description Logic ALC: syntax

Definition

Let N_C be a set of concept names and N_R be a set of relation names, $N_C \cap N_R = \emptyset$. $\mathcal{ALC}$ concept descriptions are inductively defined as:

- If $A \in N_C$, then A is an $\mathcal{ALC}$ concept description
- If C, D are $\mathcal{ALC}$ concept description, and $r \in N_R$, then the following are $\mathcal{ALC}$ concept descriptions:
  - $C \sqcap D$
  - $C \sqcup D$
  - $\neg C$
  - $\forall r.C$
  - $\exists r.C$

- Use $\bot$ as abbreviation of $A \sqcap \neg A$, $\top$ as abbreviation of $A \sqcup \neg A$

Description Logic ALC: semantics

Definition

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$:

- $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ for all $A \in N_C$,
- $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ for all $r \in N_R$

The interpretation function is extended to $\mathcal{ALC}$ concept descriptions as follows:

- $(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$
- $(C \sqcup D)^{\mathcal{I}} := C^{\mathcal{I}} \cup D^{\mathcal{I}}$
- $(\neg C)^{\mathcal{I}} := \Delta^{\mathcal{I}} - C^{\mathcal{I}}$
- $(\forall r.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} | \text{for all } e \in \Delta^{\mathcal{I}} : (d, e) \in r^{\mathcal{I}} \text{ implies } e \in C^{\mathcal{I}}\}$
- $(\exists r.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} | \text{there is } e \in \Delta^{\mathcal{I}} : (d, e) \in r^{\mathcal{I}} \text{ and } e \in C^{\mathcal{I}}\}$

Description Logic: terminologies

- A TBox axiom is of the form $A \sqsubseteq B$ (`A SubClassOf: B`) where A and B are concept descriptions
- $A \equiv B$ (`A EquivalentTo: B`) is $A \sqsubseteq B$ and $B \sqsubseteq A$
- `A DisjointWith B` is $A \sqcap B \sqsubseteq \bot$
- An interpretation $\mathcal{I}$ is a model of a TBox $\mathcal{T}$ if it satisfies all its axioms: $A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$ for all $A \subseteq B \in \mathcal{T}$

Description Logic: assertions

- An assertion is of the form $C(a)$ (concept assertion) or $r(a, b)$ (role assertion), where C is a concept description, r is a role, a, b are individual names from a set N_I of such names
- An ABox is a finite set of assertions
- An interpretation $\mathcal{I}$ is a model of an ABox $\mathcal{A}$ if it satisfies all its assertions:
  - $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for all $C(a) \in \mathcal{A}$
  - $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ for all $r(a, b) \in \mathcal{A}$

Description Logic: Reasoning

- Subsumption: Is C a subconcept of D?
  - $C \sqsubseteq_{\mathcal{T}} D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$
- Satisfiability: Is the concept C non-contradictory?
  - C is satisfiable w.r.t. $\mathcal{T}$ iff $C^{\mathcal{I}} \neq \emptyset$ for some model $\mathcal{I}$ of $\mathcal{T}$
- Consistency: Is the ABox $\mathcal{A}$ non-contradictory?
  - $\mathcal{A}$ is consistent w.r.t. $\mathcal{T}$ iff it has a model that is also a model of $\mathcal{T}$
- Instantiation: Is e an instance of C?
  - $\mathcal{A} \models_{\mathcal{T}} C(e)$ iff $e^{\mathcal{I}} \in C^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$.

Offtopic: knowledge graphs

Does this relate to knowledge graphs?

Offtopic: knowledge graphs

Does this relate to knowledge graphs?
A knowledge graph is an ABox + RBox

- ontologies are (mostly) the TBox!

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

Using background knowledge

Problem statement (first attempt):

Given a set of entities (instances) within an ontology (DL theory). Can we discover/predict *new* relations between the entities, or between entities and classes in the ontology?

Using background knowledge

Problem statement (first attempt):

Given a set of entities (instances) within an ontology (DL theory). Can we discover/predict *new* relations between the 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 entities (instances) within an ontology (DL theory). Can we discover/predict *new* relations between the 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 entities (instances) within an ontology (DL theory). Can we discover/predict *new* relations between the 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; add new classes; expand definition; constrained learning; etc.
  - computing "similarity"

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

This tutorial is about machine learning; why do I talk about semantic similarity?

Semantic similarity or machine learning

This tutorial is about machine learning; why do I talk about semantic similarity?

- 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

Semantic similarity or machine learning

This tutorial is about machine learning; why do I talk about semantic similarity?

- 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
- semantic similarity and machine learning with ontologies can have similar aims
  - predict relations; determine similarity; use background knowledge in “features”

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
  - model-based: define similarity within (special) Σ-structures

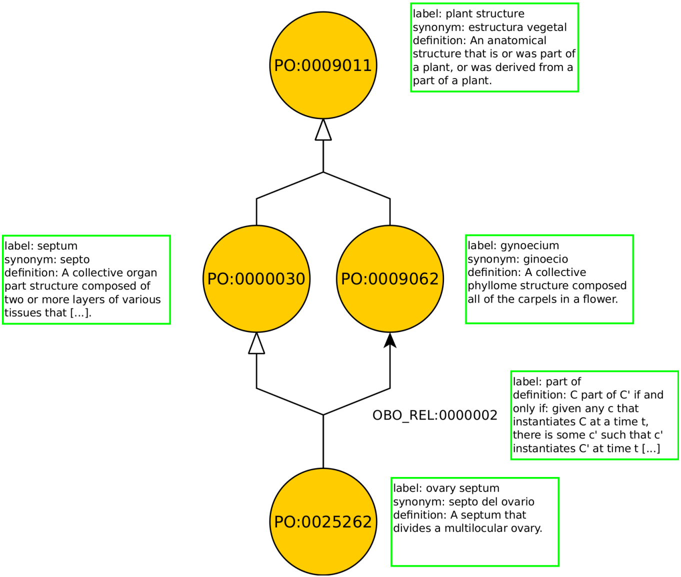
Ontologies and graphs

- 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
  - model-based: define similarity within (special) Σ-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
  - model-based: define similarity within (special) Σ-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`: $X \overset{\equiv}{\leftrightarrow} Y$, $\{X, Y\}$
- …

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`: $X \overset{\equiv}{\leftrightarrow} Y$, $\{X, Y\}$
- ...

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

Asserted and inferred

relation patterns can be asserted or inferred

- `X 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`
- Therefore: $X \xrightarrow{\text{part-of}} Z$
- $\Rightarrow$ we should use deductive inference to generate these patterns

OWL2Vec* Graph Conversion Rules

Axiom of condition 1	Axiom or triple(s) of condition 2	Projected triple(s)
$A \sqsubseteq \Box r.D$	$D \equiv B \mid B_1 \sqcup ... \sqcup B_n \mid B_1 \sqcap ... \sqcap B_n$	$\langle A, r, B \rangle$ or
or		
$\Box r.D \sqsubseteq A$		
$\exists r.\top \sqsubseteq A$ (domain)	$\top \sqsubseteq \forall r.B$ (range)	$\langle A, r, B_i \rangle$ for $i \in 1, ..., n$
$A \sqsubseteq \exists r.\{b\}$	$B(b)$	
$r \sqsubseteq r'$	$\langle A, r', B \rangle$ has been projected	
$r' \equiv r^-$	$\langle B, r', A \rangle$ has been projected	
$s_1 \circ ... \circ s_n \sqsubseteq r$	$\langle A, s_1, C_1 \rangle ... \langle C_n, s_n, B \rangle$ have been projected	
$B \sqsubseteq A$	–	$\langle B, rdfs:subClassOf, A \rangle$
		$\langle A, rdfs:subClassOf^-, B \rangle$
$A(a)$	–	$\langle a, rdf:type, A \rangle$
		$\langle A, rdf:type^-, a \rangle$
$r(a, b)$	–	$\langle a, r, b \rangle$

RDF graphs

Also possible to use the RDF syntax of OWL to generate a graph:

```
<owl:Class rdf:about="#VegetarianPizza">
<owl:equivalentClass>
  <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
     <rdf:Description rdf:about="#Pizza"/>
     <owl:Restriction>
       <owl:onProperty rdf:resource="#hasTopping"/>
        <owl:allValuesFrom>
         <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
             <rdf:Description rdf:about="#CheeseTopping"/>
             <rdf:Description rdf:about="#VegetableTopping"/>
           </owl:unionOf>
         </owl:Class>
       </owl:allValuesFrom>
     </owl:Restriction>
    </owl:intersectionOf>
  </owl:Class>
</owl:equivalentClass>
</owl:Class>
```

Methods and tools

- 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 (why?)
  - conceptual graphs?
- OBO Format represents ontologies as graphs:
  - Protege/OWLAPI: OBO export
  - OBO toolsets (e.g., ROBOT)
  - OBO Graphs: `https://github.com/geneontology/obographs`

Methods and tools

- 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 (why?)
  - conceptual graphs?
- OBO Format represents ontologies as graphs:
  - Protege/OWLAPI: OBO export
  - OBO toolsets (e.g., ROBOT)
  - OBO Graphs: `https://github.com/geneontology/obographs`
- but: a conversion of an ontologies into a graph will almost always lead to a loss of information