Concrete Names for Complex Expressions in Ontologies: A Survey of Biomedical Ontologies

Chris Kindermann & Martin G. Skjæveland

Stanford University
christian.kindermann@stanford.edu

August 31, 2023

14th International Conference on Biomedical Ontology (ICBO 2023)

Motivation

Definitions in OBO

Stanford University

Concrete Names for Complex Expressions

OBO Principle 6:

> *The ontology has textual definitions for the majority of its classes [...].*

Definitions in OBO

Stanford University

Concrete Names for Complex Expressions

OBO Principle 6:

> *The ontology has textual definitions for the majority of its classes [...].*

Recommendation:

> *Logical definitions, when present, should agree with textual definitions and vice versa.*

Logical Definitions

Stanford University

Concrete Names for Complex Expressions

A '**Liquid State**'

is a '**physical state**' that '**has the state**' of a '**liquid**'.

Logical Definitions

Stanford University

A '**Liquid State**'
is a '**physical state**' that '**has the state**' of a '**liquid**'.

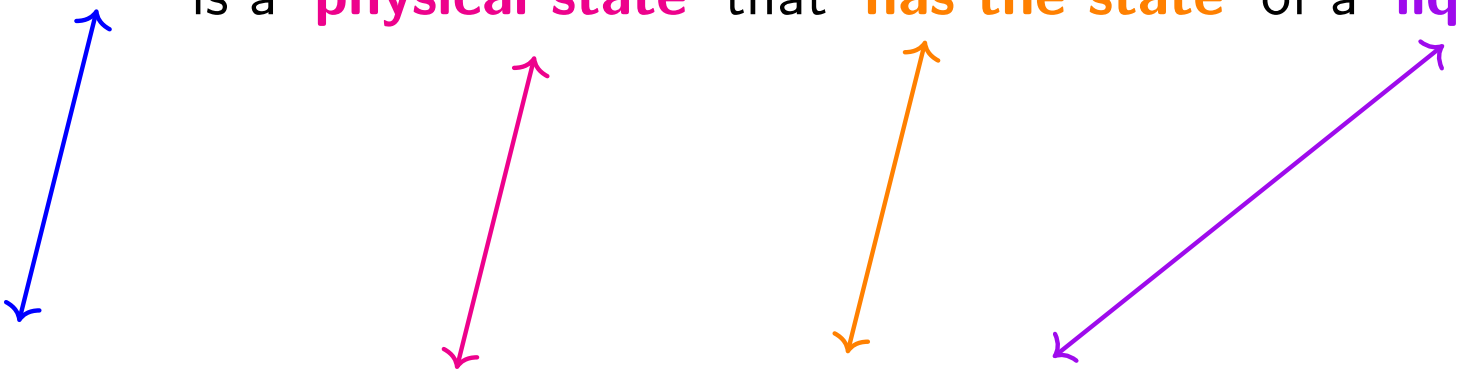
$$\mathsf{LiquidState} \equiv \mathsf{PhysicalState} \sqcap \exists\ \mathsf{hasState}.\mathsf{Liquid}$$

Logical Definitions

Stanford University

A '**Liquid State**'

is a '**physical state**' that '**has the state**' of a '**liquid**'.



$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState}.\text{Liquid}$$

Logical Definitions

Stanford University

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState}.\text{Liquid}$$

Logical Definitions

Stanford University

$$\begin{aligned}\text{Clotting} \equiv\ & \\ \exists\ & \text{actsSpecificallyOn}.(\text{Blood} \sqcap \\ & \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \\ & \exists\ \text{hasState}.\text{Liquid})) \sqcap \\ & \exists\ \text{hasOutcome}.\text{SolidBlood}\end{aligned}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState}.\text{Liquid}$$

Reusing Defined Names

Stanford University

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid})) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

Reusing Defined Names

Stanford University

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid})) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState.LiquidState}) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

Reusing Defined Names

Stanford University

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid})) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState.LiquidState}) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

(Nikitina and Koopmann 2017)

Research Questions

Research Questions

Stanford University

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid})) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

Research Questions

Stanford University

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid})) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

- How many concrete names are ***defined***?

Research Questions

Stanford University

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid})) \sqcap \exists\ \text{hasOutcome.SolidBlood}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

- How many concrete names are ***defined***?
- How often are such names ***reused***?

Research Questions

Stanford University

$$\begin{aligned}\text{Clotting} \equiv\ & \\ \exists\ & \text{actsSpecificallyOn}.(\text{Blood} \sqcap \\ & \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \\ & \exists\ \text{hasState.Liquid})) \sqcap \\ & \exists\ \text{hasOutcome.SolidBlood}\end{aligned}$$

$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState.Liquid}$$

- How many concrete names are ***defined***?
- How often are such names ***reused***?
- How often are such names ***not reused***?

Abbreviations and Synonyms

Syntactic Definitions

Stanford University

Abbreviation: named class N for complex class C

Synonym: named class N and named class S

Syntactic Definitions

Stanford University

Abbreviation: named class N for complex class C

- Simple Definition:
  EquivalentClasses(N, C)

Synonym: named class N and named class S

- Simple Definition:
  EquivalentClasses(N, S)

Syntactic Definitions

Stanford University

Abbreviation: named class N for complex class C

- Simple Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{C})$
- Ambiguous Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{C}_1, \ldots, \mathsf{C}_n)$

Synonym: named class N and named class S

- Simple Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{S})$
- Ambiguous Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{S}_1, \ldots, \mathsf{S}_n)$

Syntactic Definitions

Stanford University

Abbreviation: named class N for complex class C

- Simple Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{C})$
- Ambiguous Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{C}_1, \ldots, \mathsf{C}_n)$
- Compound Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}_1, \ldots, \mathsf{N}_m, \mathsf{C}_1, \ldots, \mathsf{C}_n)$

Synonym: named class N and named class S

- Simple Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{S})$
- Ambiguous Definition:
  $\mathit{EquivalentClasses}(\mathsf{N}, \mathsf{S}_1, \ldots, \mathsf{S}_n)$

Explicit & Implicit Definitions

Stanford University

Explicit & Implicit Definitions

Stanford University

- Explicit: $\mathit{EquivalentClasses}(\mathrm{N}, \mathrm{C}) \in \mathcal{O}$

Explicit & Implicit Definitions

Stanford University

- Explicit: $\mathit{EquivalentClasses}(\mathrm{N}, \mathrm{C}) \in \mathcal{O}$
- Implicit: $\mathcal{O} \models \mathit{EquivalentClasses}(\mathrm{N}, \mathrm{C}) \notin \mathcal{O}$

Uses and Possible Uses

Stanford University

- Given an $\mathcal{O}$

Uses and Possible Uses

Stanford University

- Given an $\mathcal{O}$
- Given (implicit & explicit) definitions *EquivalentClasses*(N, C)

Uses and Possible Uses

Stanford University

- Given an $\mathcal{O}$
- Given (implicit & explicit) definitions *EquivalentClasses*(N, C)
- Number of occurrences
  - N in $\mathcal{O}$
  - C in $\mathcal{O}$

Uses and Possible Uses

Stanford University

- Given an $\mathcal{O}$
- Given (implicit & explicit) definitions *EquivalentClasses*(N, C)
- Number of occurrences
  - N in $\mathcal{O}$ — use of N
  - C in $\mathcal{O}$ — *possible* use of N

Empirical Results

Empirical Results

Stanford University

Concrete Names for Complex Expressions

1. Syntactic Definitions

Empirical Results

Stanford University

1. Syntactic Definitions

- $309/744 \approx$ **41%** ontologies with abbreviations
- $136/744 \approx$ **18%** ontologies with synonyms

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - $309/744 \approx$ **41%** ontologies with abbreviations
   - $136/744 \approx$ **18%** ontologies with synonyms
   - cases with more than **10,000** and **100,000** definitions

Empirical Results

Stanford University

1. Syntactic Definitions

- many abbreviations defined in many ontologies

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - in **80%** of ontologies, **all** abbreviation are also used

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - in **80%** of ontologies, **all** abbreviation are also used
   - abbreviations can be used up to **1,000–10,000** times

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - many abbreviations used, sometimes frequently

Empirical Results

Stanford University

Concrete Names for Complex Expressions

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - many abbreviations used, sometimes frequently
3. **Possible Uses**

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - many abbreviations used, sometimes frequently
3. **Possible Uses**
   - *explicit*: possible uses in $101/309 \approx$ **33%** ontologies

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - many abbreviations used, sometimes frequently
3. **Possible Uses**
   - *explicit*: possible uses in $101/309 \approx$ **33%** ontologies
   - *implicit*: possible uses in **all** ontologies

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - many abbreviations used, sometimes frequently
3. **Possible Uses**
   - *explicit*: possible uses in $101/309 \approx$ **33%** ontologies
   - *implicit*: possible uses in **all** ontologies
   - cases of $\geq$ **1000** possible uses for a single abbreviation

Empirical Results

Stanford University

1. **Syntactic Definitions**
   - many abbreviations defined in many ontologies
2. **Uses**
   - many abbreviations used, sometimes frequently
3. **Possible Uses**
   - abbreviations with many possible uses exist

Conclusion

Open Questions

Should equivalent named classes . . .

Open Questions

Stanford University

Should equivalent named classes . . .

- be *reused*?

Open Questions

Stanford University

Should equivalent named classes . . .

- be *reused*?
- be *introduced* for reoccurring complex classes?

Open Questions

Stanford University

Should equivalent named classes . . .

- be *reused*?
- be *introduced* for reoccurring complex classes?
- be *non-ambiguous*?

Open Questions

Stanford University

Concrete Names for Complex Expressions

Should equivalent named classes . . .

- be *reused*?
- be *introduced* for reoccurring complex classes?
- be *non-ambiguous*?

When & Why and When & Why ***not***?

Open Questions

Stanford University

Should equivalent named classes ...

- be *reused*?
- be *introduced* for reoccurring complex classes?
- be *non-ambiguous*?

When & Why and When & Why ***not***?

Let me know:

christian.kindermann@stanford.edu

Future Work

Stanford University

- *exhaustive* reuse of abbreviations

Future Work

Stanford University

- *exhaustive* reuse of abbreviations

$$\begin{aligned}\text{Clotting} \equiv\ & \\ \exists\ & \text{actsSpecificallyOn}.(\text{Blood} \sqcap \\ & \quad \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \\ & \qquad \exists\ \text{hasState}.\text{Liquid})) \sqcap \\ & \quad \exists\ \text{hasOutcome}.\text{SolidBlood}\end{aligned}$$

Future Work

Stanford University

$$\text{LiquidBlood} \equiv \text{Blood} \sqcap \exists\ \text{hasPhysicalState}.\text{LiquidState}$$
$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState}.\text{Liquid}$$

- *exhaustive* reuse of abbreviations

$$\begin{aligned}\text{Clotting} \equiv\ & \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \\ & \exists\ \text{hasPhysicalState}.(\text{PhysicalState} \sqcap \\ & \exists\ \text{hasState}.\text{Liquid})) \sqcap \\ & \exists\ \text{hasOutcome}.\text{SolidBlood}\end{aligned}$$

Future Work

Stanford University

$$\text{LiquidBlood} \equiv \text{Blood} \sqcap \exists\ \text{hasPhysicalState}.\text{LiquidState}$$
$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState}.\text{Liquid}$$

- *exhaustive* reuse of abbreviations

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.(\text{Blood} \sqcap \exists\ \text{hasPhysicalState}.\text{LiquidState}) \sqcap \exists\ \text{hasOutcome}.\text{SolidBlood}$$

Future Work

Stanford University

$$\text{LiquidBlood} \equiv \text{Blood} \sqcap \exists\ \text{hasPhysicalState}.\text{LiquidState}$$
$$\text{LiquidState} \equiv \text{PhysicalState} \sqcap \exists\ \text{hasState}.\text{Liquid}$$

- *exhaustive* reuse of abbreviations

$$\text{Clotting} \equiv \exists\ \text{actsSpecificallyOn}.\text{LiquidBlood} \sqcap \exists\ \text{hasOutcome}.\text{SolidBlood}$$

Future Work

Stanford University

- exhaustive reuse of abbreviations
- using names *without needing new* named class

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names *without needing new* named class

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names *without needing new* named class

$\alpha =$ ItalianPizza EquivalentTo Pizza and hasCountryOfOrigin value Italy

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names *without needing new* named class

$\alpha =$ ItalianPizza EquivalentTo Pizza and hasCountryOfOrigin value Italy

$$\mathcal{O} \not\models \alpha \text{ whereas } \mathcal{O} \cup \{\alpha\} \models \alpha.$$

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names *without needing new* named class

`ItalianPizza` $\mapsto$ Pizza and hasCountryOfOrigin value Italy

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names *without needing new* named class

`ItalianPizza` $\mapsto$ Pizza and hasCountryOfOrigin value Italy

$\mathcal{O}_{\mathcal{T}} = \{$ Napoletana SubClassOf `ItalianPizza`,
Diavola SubClassOf `ItalianPizza`,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names *without needing new* named class

$$\texttt{ItalianPizza} \mapsto \text{Pizza and hasCountryOfOrigin value Italy}$$

$\mathcal{O}_T = \{$ Napoletana SubClassOf `ItalianPizza`,
Diavola SubClassOf `ItalianPizza`,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

$$\mathcal{O} \equiv \mathcal{O}_T$$

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names without needing new named class
- naming things using some notion of *abstraction*

`ItalianPizza` $\mapsto$ Pizza and hasCountryOfOrigin value Italy

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names without needing new named class
- naming things using some notion of *abstraction*

$\texttt{ItalianPizza} \mapsto$ Pizza and hasCountryOfOrigin value Italy

$\texttt{PizzaWithOrigin}(x) \mapsto$ Pizza and hasCountryOfOrigin value x

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

- exhaustive reuse of abbreviations
- using names without needing new named class
- naming things using some notion of *abstraction*

$\texttt{ItalianPizza} \mapsto$ Pizza and hasCountryOfOrigin value Italy

$\texttt{PizzaWithOrigin}(x) \mapsto$ Pizza and hasCountryOfOrigin value x

$\mathcal{O}_P = \{$ Napoletana SubClassOf $\texttt{PizzaWithOrigin}$(Italy),
Diavola SubClassOf $\texttt{PizzaWithOrigin}$(Italy),
Hawaiian SubClassOf $\texttt{PizzaWithOrigin}$(Canada) $\}$

Future Work

Stanford University

$\mathcal{O} = \{$ Napoletana SubClassOf Pizza and hasCountryOfOrigin value Italy,
Diavola SubClassOf Pizza and hasCountryOfOrigin value Italy,
Hawaiian SubClassOf Pizza and hasCountryOfOrigin value Canada$\}$

Reasonable Ontology Templates (OTTR)
(`https://ottr.xyz/`)

`ItalianPizza` $\mapsto$ Pizza and hasCountryOfOrigin value Italy

`PizzaWithOrigin`$(x) \mapsto$ Pizza and hasCountryOfOrigin value x

$\mathcal{O}_P = \{$ Napoletana SubClassOf `PizzaWithOrigin`(Italy),
Diavola SubClassOf `PizzaWithOrigin`(Italy),
Hawaiian SubClassOf `PizzaWithOrigin`(Canada) $\}$