Bioinformatics, 36(10), 2020, 3207–3214
doi: 10.1093/bioinformatics/btaa106
Advance Access Publication Date: 17 February 2020
Original Paper

OXFORD

Databases and ontologies

SSIF: Subsumption-based Sub-term Inference Framework to audit Gene Ontology

Rashmie Abeysinghe[1,2], Eugene W. Hinderer III[3], Hunter N. B. Moseley [3,4,5,6] and Licong Cui [1,*]

[1]School of Biomedical Informatics, University of Texas Health Science Center at Houston, Houston, TX 77030, USA, [2]Department of Computer Science, [3]Department of Molecular and Cellular Biochemistry, [4]Institute for Biomedical Informatics, [5]Markey Cancer Center and [6]Center for Environmental and Systems Biochemistry, University of Kentucky, Lexington, KY 40506, USA

*To whom correspondence should be addressed.

Associate Editor: Pier Luigi Martelli

Received on August 23, 2019; revised on February 8, 2020; editorial decision on February 10, 2020; accepted on February 11, 2020

Abstract

Motivation: The Gene Ontology (GO) is the unifying biological vocabulary for codifying, managing and sharing biological knowledge. Quality issues in GO, if not addressed, can cause misleading results or missed biological discoveries. Manual identification of potential quality issues in GO is a challenging and arduous task, given its growing size. We introduce an automated auditing approach for suggesting potentially missing *is-a* relations, which may further reveal erroneous *is-a* relations.

Results: We developed a Subsumption-based Sub-term Inference Framework (SSIF) by leveraging a novel term-algebra on top of a sequence-based representation of GO concepts along with three conditional rules (monotonicity, intersection and sub-concept rules). Applying SSIF to the October 3, 2018 release of GO suggested 1938 unique potentially missing *is-a* relations. Domain experts evaluated a random sample of 210 potentially missing *is-a* relations. The results showed SSIF achieved a precision of 60.61, 60.49 and 46.03% for the monotonicity, intersection and sub-concept rules, respectively.

Availability and implementation: SSIF is implemented in Java. The source code is available at https://github.com/rashmie/SSIF.

Contact: licong.cui@uth.tmc.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The Gene Ontology (GO), recognized as a tool for the unification of biology (Ashburner *et al.*, 2000), has been widely used for codifying, managing and sharing biological knowledge through the annotation of genes, gene products and sequences with semantic specificity for and across organisms. GO is considered the most comprehensive and extensively used knowledge-base relating to the functions of genes and their gene products (Gene Ontology Consortium, 2018). It contains over 44 000 concepts covering three subdomains: biological process (the broad biological system in which a gene product is involved), molecular function (the specific role a gene product has or potentially has within a biological process) and cellular component (the location or organized unit in a cell where the gene product performs its molecular function) [Francis (2013), http://www.geneontology.org/page/documentation], which are organized as three separate sub-ontologies.

Relations between GO concepts include *subtype (or is-a)*, *part of*, *has part*, *regulates*, *negatively regulates* and *positively regulates*. With regard to *subtype* relations, the three sub-ontologies of GO can be treated as separate directed acyclic graphs, with concepts as nodes and subtype relations as edges between concepts in the graphs (http://geneontology.org/docs/ontology-relations/). The *subtype* relation forms the basic hierarchical structure of GO. For example, *A is-a B* means that node *A* is a subtype of node *B*. The subtype relation is transitive, i.e. if *A is-a B* and *B is-a C*, then *A is-a C* (Dessimoz and Škunca, 2017).

Biological knowledge captured in GO is continuously evolving. GO is updated and released monthly (Gene Ontology Consortium, 2006). Such updates are an essential part of its lifecycle. In addition to keeping current with the latest biological discoveries, a major part of the updates aims to reflect efforts in improving its quality by fixing errors, inconsistencies and other potential quality issues.

Because of its fundamental role in codifying, managing and sharing biological knowledge, quality issues in GO, if not addressed, can cause misleading results or missed biological discoveries (Alterovitz *et al.*, 2006). Therefore, enhancing the quality of GO, though a challenging and arduous task, directly impacts the very foundation of data-intensive biological discovery.

Various approaches for auditing and quality assurance have been applied to biomedical terminologies including GO (Geller *et al.*, 2018; Zhu *et al.*, 2009). Most existing quality assurance approaches for GO have focused on the enrichment of concepts in order to keep pace with the rapidly evolving biological knowledge (Dutkowski *et al.*, 2013; Maere *et al.*, 2005; Peng *et al.*, 2016; Reimand *et al.*, 2007). Some works have focused on identifying inconsistencies in GO based on the lexical features of concepts, such as investigating inconsistent expression of concept terms in GO (Verspoor *et al.*, 2009) and studying the compositional structure of GO terms (Ogren *et al.*, 2003). A few studies have attempted to address quality issues of GO from a structural point of view, such as uncovering redundant relations, missing relations and erroneous relations. In one such work, Ochs *et al.* (2016) developed two kinds of high-level summary graphs called abstraction networks for auditing GO and identified groups of anomalous terms that are expected to have a higher error rate when compared to other terms. Mougin (2015) exploited reasoning over relationships in GO to identify redundant relations and leveraged compositional structure of the concept names to detect missing relations. Xing *et al.* (2016) developed an algorithm combining dynamic programming and topological sort for exhaustive detection of redundant hierarchical relations in biomedical ontologies including GO. In a previous study, we employed a lexical-based inference approach to identify missing or erroneous hierarchical relations in GO (Abeysinghe *et al.*, 2017).

Although redundant relations may be acceptable, missing relations and erroneous relations reflect modeling issues of an ontology, and impact the quality of semantically-enabled applications, such as ontology-based search engines and ontology alignment systems (Cui *et al.*, 2017a, b; Lambrix *et al.*, 2015). Missing relations may lead to valid conclusions being missed, and erroneous relations may cause invalid conclusions. For instance, in the AmiGO web application for searching and browsing the GO database (Carbon *et al.*, 2009), missing hierarchical relations directly influence the quality of the search results with valid results being missed. As an example, suppose we want to find all genes and gene products annotated to the GO concept *cellular response to inorganic substance (GO: 0071241)*; however, concept *cellular response to oxygen radical (GO: 0071450)* is currently not listed as its subtype (i.e. missing *is-a* relation). As a consequence, all the gene products which are annotated to concept *cellular response to oxygen radical (GO: 0071450)* would be missing from the search results.

However, the main limitations of existing approaches for uncovering missing and erroneous relations in GO relations are: (i) the approach only identifies problematic areas where errors may exist and the results generated need extensive manual review by domain experts to uncover the exact quality issues (Ochs *et al.*, 2016), (ii) the approach only detects missing relations (Mougin, 2015) or (iii) the approach only leverages simple lexical features neglecting sophisticated lexical features (Abeysinghe *et al.*, 2017; Mougin, 2015). In this article, we introduce a novel Subsumption-based Subterm Inference Framework (SSIF) for uncovering not only missing relations but also erroneous relations in GO. SSIF will leverage a sequence-based term-algebra to analyze sophisticated lexical features of GO concepts and pinpoint the exact locations of quality issues.

2 Materials and methods

In this work, we use the October 3, 2018 release of GO in the Web Ontology Language (OWL) format. We first parse the OWL file to extract all the concepts and *is-a* relations in GO. Then, we compute the *is-a* transitive closure to get all the direct and indirect *is-a* relations.

We develop SSIF by leveraging both the underlying hierarchical structure of GO and a novel term-algebra. SSIF contains three main components: (i) a sequence-based representation of GO concepts constructed using part-of-speech (POS) tagging, sub-concept matching and antonym tagging; (ii) a formulation of algebraic operations for the development of a term-algebra based on the sequence-based representation, that leverages subsumption-based longest subsequence alignment; and (iii) the construction of a set of conditional rules for backward subsumption inference aimed at uncovering problematic *is-a* relations in GO.

2.1 Sequence-based representation of GO concepts

Ogren *et al.* (2003) pointed out that over 65% of GO concepts (or terms) contain another GO term as a proper substring. For instance, *negative regulation of cellular protein catabolic process (GO: 1903363)* contains the term *regulation of cellular protein catabolic process (GO: 1903362)* as a proper substring. We refer to the proper substring as a sub-concept of the original concept. In addition, we consider those GO concepts containing only alphanumeric characters, constituting almost 90% of GO concepts.

In this work, we represent each GO concept with a sequence of primitive elements, where a primitive element can be a single word or a sub-concept. Given an input concept C, we denote its sequence of elements $E(C)$ as $[e_1, e_2, e_3, \ldots, e_n]$. We further annotate the elements with tags and form the corresponding sequence of tags $T(C)$, denoted as $[t_1, t_2, t_3, \ldots, t_n]$, where tag t_i corresponds to element e_i. The following three tagging processes are performed: POS tagging, sub-concept tagging and antonym tagging.

2.1.1 POS tagging

We leverage the Stanford Parser (Toutanova *et al.*, 2003) to parse and annotate the GO terms to obtain sequence-based representations with tagged annotations for concepts. For example, the concept $C =$ *negative regulation of cellular protein catabolic process (GO: 1903363)* is represented and annotated as follows:

$$E(C) = [negative, regulation, of, cellular, protein, catabolic, process],$$

$$T(C) = [JJ, NN, IN, JJ, NN, JJ, NN],$$

where *JJ*, *NN* and *IN* are the POS tags denoting adjective, noun and preposition or subordinating conjunction, respectively.

2.1.2 Sub-concept tagging

After the POS tagging, we further detect sub-concepts contained in the concepts, i.e. the proper substrings of concepts that are also GO concepts. Then, we replace the substrings corresponding to the sub-concepts with their GO identifiers. More specifically, for a concept C with sequence-based representation $E(C) = [e_1, e_2, e_3, \ldots, e_n]$ and annotation $T(C) = [t_1, t_2, t_3, \ldots, t_n]$, if substring $[e_j, e_{j+1}, \ldots e_k]$ $(1 \leq j \leq k \leq n)$ is also a GO concept S whose identifier is $I(S)$, then we update the representation as $E(C) = [e_1, e_2, \ldots, e_{j-1}, I(S), e_{k+1}, \ldots, e_n]$ and the annotation as $T(C) = [t_1, t_2, \ldots, t_{j-1}, SC, t_{k+1}, \ldots, t_n]$, where SC denotes the sub-concept tag.

For example, for the input concept $C =$ *negative regulation of cellular protein catabolic process (GO:1903363)*, there are four sub-concepts detected: *regulation of cellular protein catabolic process (GO:1903362)*, *cellular protein catabolic process (GO: 0044257)*, *protein catabolic process (GO:0030163)* and *catabolic process (GO:0009056)*. Note that, these sub-concepts are overlapping with each other (i.e. sharing at least one word in common), in which cases, we generate multiple representations for the input concept to handle the overlap. Therefore, the input concept C has four different representations (see Table 1) corresponding to the four sub-concepts detected.

Table 2 shows the sequence-based representations and tag annotations for the concept $C =$ *innate immune response activating cell surface receptor signaling pathway (GO:0002220)*, which contains the following sub-concepts: *innate immune response (GO:0045087)*, *immune response (GO:0006955)*, *cell*

Table 1. Sequence representations for concept C = *negative regulation of cellular protein catabolic process (GO:1903363)*

Sequence representation—$E(C)$	Tag annotation—$T(C)$
negative, GO:1903362	*JJ, SC*
negative, regulation, of, GO:0044257	*JJ, NN, IN, SC*
negative, regulation, of, cellular, GO:0030163	*JJ, NN, IN, JJ, SC*
negative, regulation, of, cellular, protein, GO:0009056	*JJ, NN, IN, JJ, NN, SC*

Table 2. Sequence representations for concept C = *innate immune response activating cell surface receptor signaling pathway (GO:0002220)*

Sequence representation—$E(C)$ and Tag annotation—$T(C)$
GO:0045087, activating, GO:0005623, surface, receptor, GO:0023052, pathway
SC, VBG, SC, NN, NN, SC, NN
GO:0045087, activating, GO:0009986, receptor, GO:0023052, pathway
SC, VBG, SC, NN, SC, NN
GO:0045087, activating, GO:0007166
SC, VBG, SC
innate, GO:0006955, activating, GO:0005623, surface, receptor, GO:0023052, pathway
JJ, SC, VBG, SC, NN, NN, SC, NN
innate, GO:0006955, activating, GO:0009986, receptor, GO:0023052, pathway
JJ, SC, VBG, SC, NN, SC, NN
innate, GO:0006955, activating, GO:0007166
JJ, SC, VBG, SC

(GO:0005623), *cell surface (GO:0009986)*, *signaling (GO:0023052)* and *cell surface receptor signaling pathway (GO:0007166)*. A total of six representations are generated to capture the overlaps among sub-concepts (see Table 2). For instance, since sub-concepts *innate immune response (GO:0045087)* and *immune response (GO:0006955)* are overlapping, different representations are generated to differentiate them (see the first three representations versus the last three representations in Table 2).

2.1.3 Antonym tagging

To annotate concepts involving words with antonyms, we leverage a comprehensive collection of antonym pairs provided by WordNet (https://wordnet.princeton.edu/), the most well-known lexical database for English. If there exists an element e_i of $E(C)$ belonging to the antonym collection, then we annotate e_i with the *ANT* tag in addition to its original tag. For instance, for the concept C = *negative regulation of cellular protein catabolic process (GO: 1903363)* (in Table 1), its first element *negative* involves the antonym pair (*positive*, *negative*), thus, we add the *ANT* tag for *negative* (as shown in Table 3). Note that, the *ANT* does not replace the original POS tag but rather serves as an additional tag for the element, indicating that the element *negative* is an adjective and has an antonym. We denote the antonym of element e_i as $\neg e_i$.

2.2 Algebraic operations

The sequence-based representation of GO concepts enables alignment (or matching) between concepts. We introduce a Subsumption-based Longest Common Subsequence (SLCS) alignment approach to compare concepts. First, we define a subsumption relation between sequences of elements in GO, where an element can be a word or a GO concept. Given two sequences of elements X and Y, if the term corresponding to X is a GO concept and a subtype (direct or indirect) of the term corresponding to Y, we say that X and Y have a subsumption relation, denoted as $X \preccurlyeq Y$; otherwise, we say that X and Y do not have a subsumption relation, denoted as $X \not\preccurlyeq Y$. In particular, we assume $X \preccurlyeq X$ for any sequence of elements X.

Table 3. Sequence representations for concept C = *negative regulation of cellular protein catabolic process (GO:1903363)* after antonym tagging

Sequence representation—$E(C)$	Tag annotation–$T(C)$
negative, GO:1903362	*JJ/ANT, SC*
negative, regulation, of, GO:0044257	*JJ/ANT, NN, IN, SC*
negative, regulation, of, cellular, GO:0030163	*JJ/ANT, NN, IN, JJ, SC*
negative, regulation, of, cellular, protein, GO:0009056	*JJ/ANT, NN, IN, JJ, NN, SC*

Next, we define the SLCS between two sequences of elements $X = [x_1, x_2, \ldots, x_m]$ and $Y = [y_1, y_2, \ldots, y_n]$. Let $X_i = [x_1, x_2, \ldots, x_i]$ and $Y_j = [y_1, y_2, \ldots, y_j]$ be the length i prefixes of X and length j prefixes of Y, respectively, then the SLCS between X_i and Y_j, $SLCS(X_i, Y_j)$, is defined as follows:

$$SLCS(X_i, Y_j) = \begin{cases} \varphi & \text{if } i=0 \text{ or } j=0 \\ [SLCS(X_{i-1}, Y_{j-1}), x_i] & \text{if } i,j>0 \text{ and } x_i \preccurlyeq y_j \\ [SLCS(X_{i-1}, Y_{j-1}), y_j] & \text{if } i,j>0 \text{ and } y_j \preccurlyeq x_i \\ [longest(SLCS(X_i, Y_{j-1}), SLCS(X_{i-1}, Y_j))] & \text{if } i,j>0 \text{ and } x_i \not\preccurlyeq y_j \text{ and } y_j \not\preccurlyeq x_i. \end{cases}$$

Hence, the SLCS between X and Y, $SLCS(X, Y) = SLCS(X_m, Y_n)$. For instance, consider the two concepts C_1 = *negative regulation by host of symbiont molecular function (GO:0052405)* and C_2 = *positive regulation by host of symbiont catalytic activity (GO:0043947)*, as well as their sequence representations [*negative, regulation, by, host, of, symbiont, GO:0003674*] and [*positive, regulation, by, host, of, symbiont, GO:0003824*]. Since *catalytic activity (GO:0003824)* is a subtype of *molecular function (GO:0003674)*, we have $SLCS(C_1, C_2)$ = [*regulation, by, host, of, symbiont, GO:0003824*].

The SLCS between sequences of elements allows us to define an algebraic operation *intersection* ($\sqcap$) as follows. Given two sequences of elements X and Y, there are two possible cases:

Case I: $X \preccurlyeq Y$

- In this case, we define $X \sqcap Y = X$. That is to say, if the term corresponding to X is a subtype of (or more specific than) the term corresponding to Y, then $X \sqcap Y$ is defined as the sequence of the more specific term. For example, since *catabolic process (GO:0009056)* $\preccurlyeq$ *metabolic process (GO:0008152)*, we have *catabolic process (GO:0009056)* $\sqcap$ *metabolic process (GO:0008152)* = *catabolic process (GO:0009056)*. In particular, we define $X \sqcap X = X$ for any sequence of elements X. For instance, *protein* $\sqcap$ *protein* = *protein*.

Case II: $X \not\preccurlyeq Y$

- Suppose the SLCS between two concepts $X = [x_1, x_2, \ldots, x_m]$ and $Y = [y_1, y_2, \ldots, y_n]$ is $SLCS(X, Y) = [e_1, e_2, \ldots, e_s]$, where $s \leq m$ and $s \leq n$. Then we define $X \sqcap Y$ as follows:
  1. If $s = m = n$, then $X \sqcap Y$ is defined as the sequence obtained by performing intersections between elements in X and Y, i.e.

$$X \sqcap Y = [(x_1 \sqcap y_1), (x_2 \sqcap y_2), \ldots, (x_s \sqcap y_s)] = [e_1, e_2, \ldots, e_s] = SLCS(X, Y).$$

  For instance, for X = [*cytoplasmic microtubule (GO:0005881)*, *depolymerization*] and Y = [*astral microtubule (GO:0000235)*,

depolymerization], since *astral microtubule (GO:0000235)* ≼ *cytoplasmic microtubule (GO:0005881)*, we have

$$\begin{aligned} X \sqcap Y &= [(cytoplasmic\ microtubule\ (GO:0005881) \sqcap \\ &\quad astral\ microtubule (GO:0000235)), \\ &\quad (depolymerization \sqcap depolymerization)] \\ &= [astral\ microtubule (GO:0000235), depolymerization] \\ &= Y. \end{aligned}$$

- If $s = m$ and $s < n$, then $X \sqcap Y$ is defined as the sequence obtained by replacing elements in Y with the corresponding elements in $SLCS(X, Y)$, i.e. performing intersections between elements in X and Y corresponding to those in $SLCS(X, Y)$ while keeping the remaining elements in Y intact. Take X = [*protein, catabolic process (GO:0009056)*] and Y = [*cellular, protein, metabolic process (GO:0008152)*] as an example, since *catabolic process (GO:0009056)* ≼ *metabolic process (GO:0008152)*, we have $SLCS(X, Y)$ = [*protein, catabolic process (GO: 0009056)*] and

$$\begin{aligned} X \sqcap Y &= [cellular, (protein \sqcap protein), \\ &\quad (catabolic\ process (GO:0009056) \sqcap \\ &\quad metabolic\ process (GO:0008152))] \\ &= [cellular, protein, catabolicprocess (GO:0009056)]. \end{aligned}$$

- Similarly, if $s < m$ and $s = n$, then, we define $X \sqcap Y$ as the sequence obtained by replacing elements in X with the corresponding elements in $SLCS(X, Y)$, i.e. performing intersections between elements in X and Y corresponding to those in $SLCS(X, Y)$ while keeping the remaining elements in X intact.
- In all other cases, $X \sqcap Y$ is defined as φ.

2.3 Conditional rules for backward subsumption-based inference

Based on the above-defined algebraic operations, we introduce three conditional rules for performing backward subsumption-based inference in order to identify potential problematic *is-a* relations in GO: missing *is-a* relations or erroneous *is-a* relations.

2.3.1 Monotonicity rule

Given two GO concepts A and B such that $E(A)$ and $E(B)$ have the same number of elements, $E(A) = [a_1, a_2, a_3, \ldots, a_n]$ and $E(B) = [b_1, b_2, b_3, \ldots, b_n]$. A suggestion of $A \preccurlyeq B$ or A *is-a* B (a potentially missing *is-a* relation) may be made, if the following conditions are met:

1. $a_i \preccurlyeq b_i$ holds for all i ($1 \leq i \leq n$);
2. A is currently not a subtype of B; and
3. there does not exist an element a_i in $E(A)$ with a tag *ANT* such that $\neg a_i$ is in $E(B)$.

Take two concepts A = *cellular response to oxygen radical (GO:0071450)* and B = *cellular response to inorganic substance (GO:0071241)* shown in Figure 1 as an example, where the sequence-based representations of A and B are $E(A)$ = [*cellular, response to oxygen radical (GO:0000305)*] and $E(B)$ = [*cellular, response to inorganic substance (GO:0010035)*], respectively. Since *cellular* ≼ *cellular* and *response to oxygen radical (GO:0000305)* ≼ *response to inorganic substance (GO:0010035)*, a suggestion of $A \preccurlyeq B$ may be made, i.e. *cellular response to oxygen radical (GO:0071450)* is a subtype of *cellular response to inorganic substance (GO:0071241)*.

Note that, the validity of the suggested missing *is-a* relation still need to be verified by domain experts. If the suggested missing *is-a* relation is valid, then, it is indeed a missing *is-a* relation (e.g. Fig. 1). If the suggested missing *is-a* relation is invalid, but there exists j ($1 \leq j \leq n$) such that $a_j \preccurlyeq b_j$ is an erroneous relation, which leads to the invalid suggestion, then $a_j \preccurlyeq b_j$ can be identified as an erroneous relation in GO.

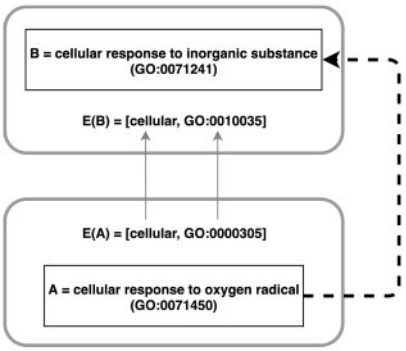

Fig. 1. An example of two GO concepts satisfying the monotonicity rule and revealing a missing *is-a* relation: *GO:0071450 is-a GO:0071241* (see the bolded, dashed arrow)

For example, in Figure 2, concept A = *pyridine nucleotide catabolic process (GO:0019364)* has a sequence-based representation $E(A)$ = [*pyridine, nucleotide catabolic process (GO:0009166)*] and concept B = *pyridine biosynthetic process (GO:0019364)* has a sequence-based representation $E(B)$ = [*pyridine, biosynthetic process (GO:0009058)*]. Since *pyridine* ≼ *pyridine* and *GO:0009166* ≼ *GO:0009058*, a suggestion of *pyridine nucleotide catabolic process (GO:0019364) is-a pyridine biosynthetic process (GO:0046220)* may be made. However, this is an invalid suggestion due to an erroneous existing *is-a* relation: *nucleotide catabolic process (GO:0009166)* ≼ *biosynthetic process (GO:0009058)*, since catabolism is not anabolism (biosynthesis).

2.3.2 Intersection rule

Suppose A, B and C are GO concepts such that $A \preccurlyeq B$ and $A \preccurlyeq C$. A suggestion of $A \preccurlyeq B \sqcap C$ (a potentially missing *is-a* relation) may be made, if the following conditions are satisfied:

1. $B \sqcap C$ is also a GO concept;
2. $B \sqcap C \preccurlyeq B$ and $B \sqcap C \preccurlyeq C$;
3. A is currently not a subtype of $B \sqcap C$; and
4. there does not exist an element a_i in $E(A)$ with a tag *ANT* such that $\neg a_i$ is in $E(B)$.

Intuitively, it is suggested that $B \sqcap C$ is the maximal concept that is more specific than both B and C.

For instance, in Figure 3, concept A = *negative regulation of ornithine catabolic process (GO:1903267)* is a subtype of concept B = *negative regulation of cellular amine metabolic process (GO:0033239)* and also a subtype of concept C = *regulation of cellular catabolic process (GO:0031329)*. $B \sqcap C$ = *negative regulation of cellular amine catabolic process (GO:0033242)* is also a GO concept, which is a subtype of A and B as well. Therefore, a suggestion of A *is-a* $B \sqcap C$ may be made, i.e. *negative regulation of ornithine catabolic process (GO:1903267)* is a subtype of *negative regulation of cellular amine catabolic process (GO:0033242)*.

If the suggested missing *is-a* relation is valid, then, it is indeed a missing *is-a* relation (e.g. Fig. 3). If the suggested missing *is-a* relation is invalid, but there exists erroneous *is-a* relation(s) among $A \preccurlyeq B$, $A \preccurlyeq C$, $B \sqcap C \preccurlyeq B$ and $B \sqcap C \preccurlyeq C$ leading to the invalid suggestion, then erroneous *is-a* relation(s) in GO can be identified.

For example, in Figure 4, concept A = *positive regulation of B cell deletion (GO:0002869)* is a subtype of concept B = *regulation of acute inflammatory response (GO:0002673)* and also a subtype of concept C = *positive regulation of biological process (GO:0048518)*. $B \sqcap C$ = *positive regulation of acute inflammatory response (GO:0002675)* is also a GO concept, which is a subtype of A and B as well. Therefore a suggestion of A *is-a* $B \sqcap C$ may be made, i.e. *positive regulation of B cell deletion (GO:0002869)* is a

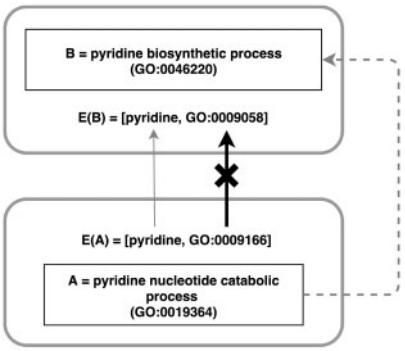

Fig. 2. An example of two GO concepts satisfying the monotonicity rule and revealing an erroneous *is-a* relation: *nucleotide catabolic process (GO:0009166) is-a biosynthetic process (GO:0009058)* (see the bolded arrow with a cross)

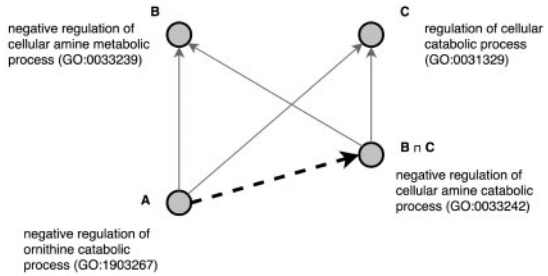

Fig. 3. An example of four GO concepts satisfying the intersection rule and revealing a missing *is-a* relation: *negative regulation of ornithine catabolic process (GO:1903267)* is a subtype of *negative regulation of cellular amine catabolic process (GO:0033242)* (see the bolded, dashed arrow)

subtype of *positive regulation of acute inflammatory response (GO:0002675)*. However, this is an invalid suggestion due to an erroneous existing *is-a* relation: *positive regulation of B cell deletion (GO:0002869) is-a regulation of acute inflammatory response (GO:0002673)*. The main purpose of B cell deletion is to produce immune tolerance. Since tolerance induction is a long process (not something that is acute), it is incorrect that *positive regulation of B cell deletion (GO:0002869)* is a subtype of *regulation of acute inflammatory response (GO:0002673)*.

2.3.3 Sub-concept rule

Given a concept C with a sequence-based representation as $E(C) = [e_1, e_2, e_3, \ldots, e_{n-1}, e_n]$ and a tag annotation as $T(C) = [t_1, t_2, t_3, \ldots, t_{n-1}, t_n]$. A suggestion of $C \preccurlyeq e_n$ (a potentially missing *is-a* relation) may be made, if the following conditions are met:

1. $t_n = SC$, i.e. the last element e_n is also a GO concept;
2. $t_i \in \{NN, JJ, SC\}$ for each i $(1 \leq i \leq n-1)$, i.e. the tags $t_1, t_2, t_3, \ldots, t_{n-1}$ are either noun, adjective or sub-concept;
3. C is currently not a subtype of e_n; and
4. there does not exist an element a_i in $E(C)$ with a tag *ANT* such that $\neg a_i$ is in e_n.

For instance, concept C = *nerve growth factor receptor binding (GO:0005163)* has a sequence-based representation $E(C) =$ [*nerve, growth factor receptor binding (GO:0070851)*] with a tag annotation $T(C) =$ [*NN, SC*]. Since the last element *growth factor receptor binding (GO:0070851)* is also a GO concept and the remaining element *nerve* is a noun, a suggestion of *nerve growth factor receptor binding (GO:0005163) is-a growth factor receptor binding (GO:0070851)* may be made.

If the suggested missing *is-a* relation is valid, then, it is indeed a missing *is-a* relation. Note that, the sub-concept rule does not leverage any existing *is-a* relation to make suggestions, thus, it cannot reveal erroneous existing *is-a* relations in GO.

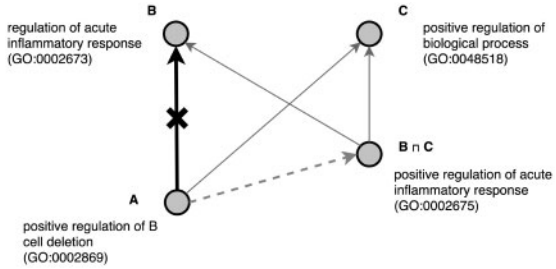

Fig. 4. An example of four GO concepts satisfying the intersection rule and revealing an erroneous existing relation: *positive regulation of B cell deletion (GO:0002869) is-a regulation of acute inflammatory response (GO:0002673) (see the bolded arrow with a cross)*

2.4 Evaluation

A random sample of potentially missing *is-a* relations is selected and evaluated by two domain experts (authors EWH and HNBM). The evaluation is performed independently by each domain expert and the disagreements between the two experts are resolved by discussion. For the monotonicity rule and intersection rule, domain experts are also provided with the existing *is-a* relations in GO that are leveraged to suggest the potentially missing *is-a* relations.

The validity of each suggested missing *is-a* relation in the random sample is evaluated by the domain experts. If the suggested missing *is-a* relation is valid, then, it is indeed a missing *is-a* relation and considered as a true positive; if the suggested missing *is-a* relation is invalid due to existing erroneous relation(s), then the erroneous *is-a* relation(s) are identified as valid and considered as true positive(s); and all the other cases are considered as false positives. The precision of SSIF according to each rule can be calculated by dividing the number of true positives by the total number of true positives and false positives.

3 Results

3.1 Summary results

For the October 3, 2018 release of GO, a total of 40 030 (out of 44 942) concepts were annotated with sequence-based representation. Among these, 30 086 concepts involve sub-concepts and 13 163 involve antonyms. The number of potentially missing *is-a* relations suggested by each conditional rule can be found in Table 4. In total, three conditional rules suggested 1938 unique potentially missing *is-a* relations. The monotonicity and intersection rules leveraged 2436 existing *is-a* relations to make these suggestions. Note that certain potentially missing *is-a* relations can be obtained by multiple rules. For instance, 11 potentially missing *is-a* relations can be obtained by both the sub-concept rule and monotonicity rule; 228 can be obtained by the monotonicity rule and intersection rule; and 1 can be obtained by all the three conditional rules.

3.2 Evaluation results

A total of 210 potentially missing *is-a* relations were randomly selected and evaluated by domain experts. Table 5 shows the number of potentially missing *is-a* relations (column 2) in the evaluation sample for each condition rule, as well as the number of valid missing *is-a* relations (column 3), the number of valid erroneous *is-a* relations (column 4), the total number of valid problematic (including both missing and erroneous) *is-a* relations (column 5) and the precision of our SSIF for identifying valid problematic *is-a* relations (column 6). For example, for the monotonicity rule, there were 99 potentially missing *is-a* obtained; 54 out of 99 were validated as missing *is-a* relations, and 6 out of 99 revealed erroneous *is-a* relations; since the total number of valid problematic *is-a* relations is 60, the precision of SSIF according to the monotonicity rule is

60.61% (= 60/99). The precisions according to the intersection rule and sub-concept rule are 60.49% (= 49/81) and 46.03% (= 29/63), respectively.

Among the evaluation sample, two potentially missing *is-a* relations were obtained by both the sub-concept rule and monotonicity rule, and were indeed missing *is-a* relations validated by domain experts; 29 potentially missing *is-a* relations were obtained by both the monotonicity rule and intersection rule, and 13 of them were validated as missing *is-a* relations and 1 of them revealed an erroneous *is-a* relation; 1 potentially missing *is-a* relation was obtained by all the 3 rules and it was validated as a missing *is-a* relation. A majority of the valid problematic *is-a* relations identified by the monotonicity rule (54 out of 60) and intersection rule (44 out of 49) are missing *is-a* relations. In sum, 120 valid problematic *is-a* relations were verified by domain experts, including 110 missing *is-a* relations and 10 erroneous *is-a* relations.

Table 6 lists 10 examples of valid problematic *is-a* relations in the evaluation sample verified by domain experts, including both missing and erroneous *is-a* relations. For instance, the first example shows a missing *is-a* relation obtained by the monotonicity rule: *cellular response to ketone (GO:1901655)* is a subtype of *cellular response to organic substance (GO:0071310)*. A complete list of missing *is-a* relations and erroneous *is-a* relations can be found in the Supplementary Material 'Missing.xlsx' and 'Erroneous.xlsx', respectively.

The valid problematic *is-a* relations indicate that the logical definitions of GO concepts could be further improved. For a valid missing *is-a* relation, it could be added to the logical definition of its corresponding sub-concept. For example, the relation *positive regulation of actin filament annealing (GO:0110056) is-a positive regulation of cytoskeleton organization (GO:0051495)* can be directly added to the logical definition of the sub-concept *positive regulation of actin filament annealing (GO:0110056)*. For a valid erroneous *is-a* relation, if the sub- and super-concepts have a direct *is-a* relation, then, the *is-a* relation can be directly removed from the logical definition of the sub-concept; if the sub- and super-concepts have an indirect *is-a* relation, then further investigation is needed to find out the root cause and make an appropriate correction.

4 Discussion

4.1 Evaluation metrics

In this article, we focused on evaluating the performance of SSIF in terms of the *precision*, which was calculated by dividing the number of true positives by the total number of true positives and false positives in the evaluation sample. Note that, unlike traditional classification tasks, it is infeasible to measure actual *recall* due to the discovery nature of the quality assurance task, i.e. there is lack of reference standard (or ground truth) that contains false negatives for calculating the recall.

However, one may use cumulative GO changes over different versions as a surrogate standard for evaluating *retrospective recall* as introduced in Zhang *et al.* (2017). For instance, we applied SSIF on the October 3, 2018 release of GO, which contained an erroneous *is-a* relation: *glucose catabolic process to lactate via pyruvate (GO:0019661) is-a pyridine nucleotide metabolic process (GO:0019362)*; this relation has been corrected and no longer exists in the current version. Such changes may serve as a partial reference standard to compute the retrospective recall.

As an experiment, we compared the October 7, 2019 release and October 3, 2018 release of GO to create a partial reference standard. There were 1886 direct *is-a* relations, which were newly added in the October 7, 2019 release. Among these, 991 were due to the introduction of new concepts; 348 were already existent as indirect *is-a* relations in the October 3, 2018 release; and 107 involved concepts, which were not used in this work since they contained non-alphanumeric characters. Therefore, we consider the remaining 440 newly added relations in the October 7, 2019 release as the partial reference standard for missing *is-a* relations. Similarly, there were 3988 direct *is-a* relations, which were removed from the October 3, 2018 release. Among these, 3049 were due to concepts, which were either replaced or made obsolete; 370 were indirect *is-a* relations in the October 7, 2019 release; 71 involved concepts, which contained non-alphanumeric characters. Therefore, we consider the remaining 498 removed relations as the partial reference standard for erroneous *is-a* relations.

Among the potentially missing *is-a* relations suggested by our SSIF, 46 were contained in the partial reference standard. Among the existing *is-a* relations, which were leveraged by SSIF to suggest potentially missing *is-a* relations, 27 were contained in the partial reference standard. As a result, SSIF achieved a retrospective recall of 7.78%, i.e. (46 + 27)/(440 + 498). In addition, 10 potentially missing *is-a* relations suggested by SSIF were indirect *is-a* relations in the October 7, 2019 release, indicating that they are also valid suggestions; and 42 indirect *is-a* relations in the October 3, 2018 release no longer exist in the October 7, 2019 release, indicating that they are erroneous *is-a* relations.

The low value of the retrospective recall is expected since it is calculated purely based on a partial reference standard obtained through version differences. The actual recall should be higher than the retrospective recall, which can be seen from the fact that in the October 3, 2018 release of GO, only 6 out of 110 valid missing *is-a* relations verified by domain experts were reflected in the October 7, 2019 release, and only 2 out of 10 erroneous *is-a* relations were removed in the October 7, 2019 release. We will submit these verified suggestions to the GO Consortium for consideration of including them in future releases of GO.

4.2 Distinction with OWL reasoners

OWL reasoners, such as ELK (Kazakov *et al.*, 2014) and Arachne (Balhoff *et al.*, 2018), are used to check the consistency of GO, and to infer implicit knowledge from explicitly stated facts and axioms. The inference typically involves the reclassification of individuals to new classes (or concepts), and classes to new super-classes, depending on their stated relations. In other words, OWL reasoners infer additional *is-a* relations based on the stated *is-a* relations.

Our SSIF approach is designed for the inferred version of GO where an OWL reasoner has already been applied to obtain

Table 4. Number of potentially missing *is-a* relations suggested by each conditional rule

Conditional rule	No. of potentially missing *is-a*
Monotonicity rule	819
Intersection rule	691
Sub-concept rule	669

Table 5. The numbers of potentially missing *is-a* relations, valid missing *is-a* relations, valid erroneous *is-a* relations, valid problematic *is-relations* respectively in the evaluation sample for each condition rule

Conditional rule	No. of potentially missing *is-a*	No. of valid missing *is-a*	No. of valid erroneous *is-a*	Total no. of valid problematic *is-a*	Precision (%)
Monotonicity rule	99	54	6	60	60.61
Intersection rule	81	44	5	49	60.49
Sub-concept rule	63	29	N/A	29	46.03

Table 6. Examples of valid problematic (missing or erroneous) *is-a* relations verified by domain experts

Conditional rule	Problematic *is-a* relation	Type
Monotonicity rule	*cellular response to ketone (GO:1901655) is-a cellular response to organic substance (GO:0071310)*	Missing
Monotonicity rule	*positive regulation of actin filament annealing (GO:0110056) is-a positive regulation of cytoskeleton organization (GO:0051495)*	Missing
Monotonicity rule	*endoplasmic reticulum membrane (GO:0005789) is-a organelle membrane (GO:0031090)*	Missing
Monotonicity rule	*cytosolic oxoglutarate dehydrogenase complex (GO:0045248) is-a cytosolic tricarboxylic acid cycle enzyme complex (GO:0045246)*	Missing
Monotonicity rule	*regulation of sphingolipid biosynthetic process (GO:0090153) is-a regulation of macromolecule biosynthetic process (GO:0010556)*	Erroneous
Intersection rule	*pantothenate catabolic process (GO:0015941) is-a cellular amide catabolic process (GO:0043605)*	Missing
Intersection rule	*sulfolipid biosynthetic process (GO:0046506) is-a cellular lipid biosynthetic process (GO:0097384)*	Missing
Intersection rule	*glucose catabolic process to lactate via pyruvate (GO:0019661) is-a pyridine nucleotide metabolic process (GO:0019362)*	Erroneous
Sub-concept rule	*perinuclear endoplasmic reticulum membrane (GO:1990578) is-a endoplasmic reticulum membrane (GO:0005789)*	Missing
Sub-concept rule	*skeletal muscle cell differentiation (GO:0035914) is-a muscle cell differentiation (GO:0042692)*	Missing

additional *is-a* relations. SSIF aims at identifying problematic *is-a* relations that even OWL reasoners have missed. Therefore, SSIF complements OWL reasoners to enhance the completeness and soundness of the ontology by identifying potentially missing and erroneous *is-a* relations.

4.3 Analysis of false positives

Although SSIF was capable of uncovering problematic *is-a* relations in GO, it cannot completely avoid false positives. In other words, there are invalid suggestions made by SSIF. For example, the sub-concept rule suggested *nuclear membrane mitotic spindle pole body tethering complex (GO:0106084)* is a subtype of *tethering complex (GO:0099023)*. However, this relation is invalid, since *tethering complex* is defined as a complex that plays a role in vesicle tethering, while *nuclear membrane mitotic spindle pole body tethering complex* is tethering non-vesicle cellular components. Note that, *tethering complex* has been renamed as *vesicle tethering complex* in the current release of GO, in which case SSIF will not make the invalid suggestion of *GO:0106084 is-a GO:0099023*.

The monotonicity rule suggested *negative regulation of renal output by angiotensin (GO:0003083) is-a negative regulation of systemic arterial blood pressure (GO:0003085)*. This is an invalid *is-a* relation, because *negative regulation of renal output by angiotensin (GO:0003083)* is actually a subtype of *positive regulation of systemic arterial blood pressure (GO:0003084)*. Although this invalid *is-a* relation was obtained by an existing *is-a* relation: *regulation of renal output by angiotensin (GO:0002019)* is a subtype of *regulation of systemic arterial blood pressure (GO:0003073)*, the latter relation is valid as the two concepts do not specify a qualifier of positive or negative.

The intersection rule suggested *peptide cross-linking via an oxazole or thiazole (GO:0018157) is-a cellular macromolecule biosynthetic process (GO:0034645)*. This potentially missing *is-a* relation was obtained by two existing *is-a* relations: *peptide cross-linking via an oxazole or thiazole (GO:0018157) is-a cellular macromolecule metabolic process (GO:0044260)* and *peptide cross-linking via an oxazole or thiazole (GO:0018157) is-a cellular biosynthetic process (GO:0044249)*. Since biosynthesis is for the oxazole or thiazole, but not for the macromolecule (which is simply being modified), the former relation is invalid while the latter two existing relations are valid. A complete list of false positives can be found in the Supplementary Material 'FalsePositives.xlsx'.

As can be seen from Table 5, the precision of SSIF according to the sub-concept rule is lower than that of the monotonicity rule and intersection rule. Through manual review of the false positives obtained by the sub-concept rule, we found that there were 11 of the suggested potentially missing *is-a* relations, which already have a *part-of* relation in GO. For instance, the sub-concept suggested *basal plasma membrane (GO:0009925) is-a plasma membrane (GO:0005886)*, however, the two concepts already have a *part-of* relation.

4.4 Limitations and future work

A limitation of this work is that we only focused on suggesting problematic *is-a* relations in GO. As mentioned earlier, the sub-concept rule suggested some invalid *is-a* relations, which already have a *part-of* relation. We plan to further investigate other types of problematic relations in GO including *part-of*. Regarding the identification of erroneous *is-a* relations in terms of the monotonicity rule and intersection rule, although SSIF requires significantly less manual effort from domain experts than most other ontology auditing approaches (by providing rationales for the suggestions of problematic *is-a* relations), domain experts still need to review the provided existing *is-a* relations that were leveraged to make the suggestion and determine if there is any erroneous relation(s) can be identified or the original suggestion is a false positive. It would be desirable to develop an automated approach that can directly detect erroneous *is-a* relations to further reduce domain experts' manual review effort.

5 Conclusion

In this article, we introduced SSIF to identify problematic *is-a* relations in GO. SSIF models GO concepts in a sequence-based representation, formulates a term-algebra and leverages three conditional rules to perform backward subsumption inference, in order to automatically suggest potentially missing *is-a* relations, which may further reveal erroneous *is-a* relations. SSIF achieved a precision of 60.61% according to the monotonicity rule, 60.49% according to the intersection rule and 46.03% according to the sub-concept rule. Since SSIF leverages the hierarchical structure and the features of concept names, which are inherent and fundamental to biomedical terminologies, it is generally applicable to audit other biomedical terminologies.

Funding

This work was supported by the National Science Foundation (NSF) through grants [1657306, 1931134, 1419282]; and the National Institutes of Health (NIH) through grants [R21CA231904, UL1TR001998-01].

Conflict of Interest: none declared.

References

Abeysinghe,R. *et al*. (2017) Auditing subtype inconsistencies among gene ontology concepts. In: *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, *Kansas City, Missouri*. pp. 1242–1245. IEEE.

Alterovitz,G. *et al*. (2006) Go PaD: the gene ontology partition database. *Nucleic Acids Res*., **35** (**Suppl. 1**), D322–D327.

Ashburner,M. *et al*. (2000) Gene ontology: tool for the unification of biology. *Nat. Genet*., **25**, 25–29.

Balhoff,J.P. *et al*. (2018) Arachne: an OWL RL reasoner applied to gene ontology causal activity models (and beyond). In: *International Semantic Web Conference (P&D/Industry/BlueSky), Monterey, California*.

Carbon,S. *et al*.; the AmiGO Hub. (2009) AmiGO: online access to ontology and annotation data. *Bioinformatics*, **25**, 288–289.

Cui,L. *et al*. (2017a) Auditing SNOMED CT hierarchical relations based on lexical features of concepts in non-lattice subgraphs. *J. Biomed. Inform*., **78**, 177–184.

Cui,L. *et al*. (2017b) Mining non-lattice subgraphs for detecting missing hierarchical relations and concepts in SNOMED CT. *J. Am. Med. Inform. Assoc*., **24**, 788–798.

Dessimoz,C. and Škunca,N. (2017) *The Gene Ontology Handbook*. Humana Press, New York, NY, USA.

Dutkowski,J. *et al*. (2013) A gene ontology inferred from molecular networks. *Nat. Biotechnol*., **31**, 38–45.

Francis,R.W. (2013) GOLink: finding cooccurring terms across gene ontology namespaces. *Int. J. Genomics*, **2013**, 1–10.

Geller,J. *et al*. (2018) Quality assurance of biomedical terminologies and ontologies. *J. Biomed. Inform*., **86**, 106–108.

Gene Ontology Consortium (2006) The gene ontology (GO) project in 2006. *Nucleic Acids Res*., **34** (**Suppl. 1**), D322–D326.

Gene Ontology Consortium (2018) The gene ontology resource: 20 years and still going strong. *Nucleic Acids Res*., **47**, D330–D338.

Kazakov,Y. *et al*. (2014) The incredible ELK. *J. Autom. Reason*., **53**, 1–61.

Lambrix,P. *et al*. (2015) Completing the is-a structure in light-weight ontologies. *J. Biomed. Semantics*, **6**, 12.

Maere,S. *et al*. (2005) BiNGO: a Cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks. *Bioinformatics*, **21**, 3448–3449.

Mougin,F. (2015) Identifying redundant and missing relations in the gene ontology. *Stud. Health Technol. Inform*., **210**, 195–199.

Ochs,C. *et al*. (2016) Quality assurance of the gene ontology using abstraction networks. *J. Bioinform. Comput. Biol*., **14**, 1642001.

Ogren,P.V. *et al*. (2003) The compositional structure of Gene Ontology terms. *Pac. Symp. Biocomput*., **2004**, 214–225.

Peng,J. *et al*. (2016) Extending gene ontology with gene association networks. *Bioinformatics*, **32**, 1185–1194.

Reimand,J. *et al*. (2007) g:Profiler - a web-based toolset for functional profiling of gene lists from large-scale experiments. *Nucleic Acids Res*., **35** (**Suppl. 2**), W193–W200.

Toutanova,K. *et al*. (2003) Feature-rich part-of-speech tagging with a cyclic dependency network. In: *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, *Edmonton, Alberta, Canada*. pp. 173–180. Association for Computational Linguistics.

Verspoor,K. *et al*. (2009) Ontology quality assurance through analysis of term transformations. *Bioinformatics*, **25**, i77–i84.

Xing,G. *et al*. (2016) FEDRR: fast, exhaustive detection of redundant hierarchical relations for quality improvement of large biomedical ontologies. *BioData Min*., **9**, 31.

Zhang,G.-Q. *et al*. (2017) Can SNOMED CT changes be used as a surrogate standard for evaluating the performance of its auditing methods? *AMIA Annu. Symp. Proc*., **2017**, 1903.

Zhu,X. *et al*. (2009) A review of auditing methods applied to the content of controlled biomedical terminologies. *J. Biomed. Inform*., 413–425.