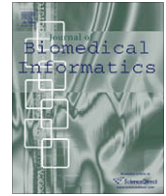




Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Relationship auditing of the FMA ontology

Huanying (Helen) Gu^{a,*}, Duo Wei^b, Jose L.V. Mejino Jr.^c, Gai Elhanan^d^a Department of Health Informatics, SHRP, University of Medicine and Dentistry of New Jersey, 65 Bergen Street, Newark, NJ 07107, USA^b CS Department, New Jersey Institute of Technology, Newark, NJ 07102, USA^c Structural Informatics Group, Department of Biological Structure, University of Washington, Seattle, WA 98195, USA^d 3M Health Information systems, Inc., Rockleigh, NJ 07647, USA

ARTICLE INFO

Article history:

Received 24 July 2008

Available online 23 January 2009

Keywords:

Ontology

FMA

Auditing

Relationship

Anatomy

ABSTRACT

The Foundational Model of Anatomy (FMA) ontology is a domain reference ontology based on a disciplined modeling approach. Due to its large size, semantic complexity and manual data entry process, errors and inconsistencies are unavoidable and might remain within the FMA structure without detection. In this paper, we present computable methods to highlight candidate concepts for various relationship assignment errors. The process starts with locating structures formed by transitive structural relationships (*part_of*, *tributary_of*, *branch_of*) and examine their assignments in the context of the IS-A hierarchy. The algorithms were designed to detect five major categories of possible incorrect relationship assignments: circular, mutually exclusive, redundant, inconsistent, and missed entries. A domain expert reviewed samples of these presumptive errors to confirm the findings. Seven thousand and fifty-two presumptive errors were detected, the largest proportion related to *part_of* relationship assignments. The results highlight the fact that errors are unavoidable in complex ontologies and that well designed algorithms can help domain experts to focus on concepts with high likelihood of errors and maximize their effort to ensure consistency and reliability. In the future similar methods might be integrated with data entry processes to offer real-time error detection.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

The Foundational Model of Anatomy (FMA) ontology is an evolving reference ontology for biomedical informatics; it is concerned with the representation of classes of entities and relationships necessary for the symbolic modeling of the phenotypic structure of the human body in a computable form that is also understandable by humans [1–3]. Specifically, the FMA explicitly represents a coherent body of declarative knowledge about human anatomy as a domain ontology. However its ontological framework can be applied and extended to other species as well. The FMA has been developed and maintained by the Structural Informatics Group at the University of Washington.

The FMA has been implemented through a *disciplined modeling approach* (as discussed in the following section). It provides comprehensive and detailed information and services for knowledge retrieval, biomedical research, anatomical education and clinical practice. The FMA currently includes more than 75,000 distinct anatomical classes and more than 200 types of relationships creating a complex semantic structure.

We hypothesize that since most of data entry in the FMA is carried out manually, it is to be expected that errors and inconsistencies might be introduced during the development of such a large

terminology. The most opportune time to detect and correct such issues is at data entry time. However, due to the large size and semantic complexity of the FMA this is not a trivial task. As a result, many errors and inconsistencies may go undetected at entry time and may remain hidden within the structure of the FMA.

Anatomy is fundamental to all biomedical domains and the FMA is most likely to be reused extensively and serve as a reference ontology for other medical knowledge sources and any undetected errors might be propagated to the recipient applications. Therefore, ensuring consistency in the FMA by well-defined auditing processes is an important and essential quality control measure for establishing the reliability and accuracy of its domain representation.

Auditing is part of ontology design life cycle [4]. In [5–9], various techniques have been presented and used to identify classification errors, redundant and circular hierarchical relationships in the UMLS [10]. Auditing methods proposed in [11–14] have been applied to SNOMED CT to ensure its consistency. Other approaches suggested in [15–18] have been used to audit large ontologies such as the FMA and NCIT.

In this paper, we describe computable methods to validate and audit the FMA *post factum* with respect to the FMA specifications and ontology modeling principles. We examine four major structural relationship types of the FMA: the *subclass*¹ and its *superclass*

¹ Relationships are written in italics.

* Corresponding author. Tel.: +1 9739720995

E-mail address: guhy@umdnj.edu (H. Gu).

inverse relationship type (IS-A) which forms the backbone hierarchy of the FMA as well as *part* and *part_of*, *branch* and *branch_of*, and *tributary* and *tributary_of*. All of the above relationship types have the transitivity property. That means that if X is a *part_of* Y and Y is a *part_of* Z, then X is a *part_of* Z. The same property holds for the other relationship types as well.

Our auditing methods examine the hierarchies formed by each of the four types of relationships as well as the interplay between the IS-A hierarchy of the FMA and each of the other three relationship types to locate any structures that may indicate incorrect or inconsistent relationship assignments. In this study, presumptive errors are defined in terms of (1) “circular relationship assignments” where a class is both an ancestor and a descendant to another class, (2) “mutually exclusive relationship assignments” where mutually exclusive relationships exist between two classes, (3) “redundant relationships assignments” where the relationship are unnecessary because they can be inferred through transitivity, (4) “inconsistent relationship assignments” where relationships are incorrectly applied at mixed levels of structural granularity and (5) “missed relationship assignments”, where expected relationships are missing. Random samples of presumptive errors were submitted for review and analysis by an FMA domain expert (J.L.V.M) to validate their applicability and to assess the efficiency of the methods used in this study.

2. Details of the FMA

The FMA currently contains over 75,000 distinct anatomical types—representing structures ranging in size from macromolecular complexes and cell components to major body parts and the whole organism itself. These types are associated with more than 135,000 terms either as preferred names, synonyms or Non-English equivalents. FMA types or classes are linked to one another by over 2.1 million relationships based on over 200 types of spatio-structural and non-structural relationships. This large and complex model is implemented using a set of declared foundational principles which provides a high-level scheme for representing the referents of classes of entities and relationships in the anatomy domain, Aristotelian definitions and a knowledge modeling environment that assures the inheritance of definitional and non-definitional attributes in a taxonomy [19]. The *Anatomy Taxonomy* (**AT**) forms the backbone IS-A hierarchy of the FMA while the network of spatio-structural relationships that exist among entities in **AT** are embodied in the Anatomical Structural Abstraction (**ASA**) component of the FMA. Together they serve to create the complex semantic structure of the FMA.

Explicit Aristotelian definitions in the **AT** consist of a class subsumption hierarchy where classes assure that general as well as more and more specific attributes that are shared by increasingly specialized anatomical structures are propagated from the root of the taxonomy to its leaves [20]. Restricting the representation to a single structural context allows for a more manageable single inheritance hierarchy; each anatomical entity except the root entity has one and only one parent in the IS-A hierarchy. However multiple links to different parent classes are allowed for the other structural relationships such as *part_of*, *branch_of* and *tributary_of* relationship types.

The FMA ontology is implemented in Protégé, a frame-based authoring environment [21–22]. A frame is a data structure that contains all the information in the ontology about a given **AT** class. This information includes the attributes of the entity to which that class refers and also the relationships of that entity to other entities. Both attributes and relationships associated with a class are expressed as slots of the frame. Non-structural attributes of FMA classes include preferred name, synonyms or its definition, while relationships are structural properties such as *has_dimension* or *bounded_by*. Non-

structural slots do not play a role in our methodology and therefore any further reference to slots assumes structural relationships.

A Protégé frame may represent a class or an instance. While a class can potentially subsume collections of more granular entities an instance is a concrete, real world object of an **AT** class, e.g., such as **John's Heart**² is an instance of the **AT** class **Heart**. Unless otherwise specified all further reference of **AT** concepts including leaf-node concepts will assume they are classes.

Facets are utilized in Protégé to impose various constraints on slots and their values that are based on the FMA rules and principles. Facets define the range of classes that can be assigned as values to a slot, the number of values that can populate a specific slot or if a slot is of a Boolean type. For example, the slot *has_part* for the class **Heart** has the facet **AT** class **Organ part** and therefore only organ parts such as the atria and ventricles which are subclasses of **Organ part** are acceptable as slot values.

An important feature of the FMA as modeled in Protégé is the ability to apply the process of inheritance to the set of template slots that a class possesses as well as own slot values. In the FMA, classes propagate downward their attributes and properties as a set of slots but their instances do not, and just as the **AT** becomes increasingly specialized as the distance from the root increases, slot attributes may follow a similar process. As opposed to a minimal approach of frame representation where attributes and values are only explicitly defined at their introduction concept and elsewhere are implied, the FMA explicitly populates values at every level. Inherited attributes and values are explicitly defined for each frame. For example, the attribute *has_dimension* is introduced at the level of **Physical anatomical entity**, in order to distinguish it from its sibling class **Non-physical anatomical entity** which does not possess any dimensional attribute. All subclasses of **Physical anatomical entity** inherit the *has_dimension* attribute which is then propagated all the way down to the leaves.

3. Methods

Our auditing methods were designed to specifically audit large ontologies such as the FMA. We applied our methodology to examine four types of relationships, *subclass*, *part_of*, *branch_of*, and *tributary_of* to determine whether the relationship assignments are in accord with the principles as declared by the FMA for the representation of these anatomical entities. Our auditing process used functions and the application programming interface available through the FMA representation in Protégé.

Before describing our auditing methods, we give the following definitions:

- *Subtype_of* (or *subclass*) is an *is_a* relationship which asserts the instantiation of a broader type by two or more narrower (more specific) types (subtypes). The inverse relationship is *has_subtype* or *has_subclass*.
- *Part_of* relationship is a structural anatomical relationship which holds between each entity of type **A** and some anatomical entity of the same dimension of type **B** such that if **A** is *part_of* **B**, there is a complement **C** which together with **A** accounts for the whole (100%) of **B**. The inverse relationship is *has_part*.
- *Regional_part_of* is a *part_of* relationship which holds between each maximally connected anatomical structure and its part demarcated from the complement by a predominantly fiat boundary.
- *Branch_of* is a *regional_part_of* relationship which holds between trunks and subtrees of a tree organ that is either tracheobronchial or biliary or of arterial or neural type.

² Classes are written in bold.

- *Tributary_of* is a *regional_part_of* relationship which holds between trunks and subtrees of a tree organ that is either of venous or lymphatic type.

The first step in the auditing process is to identify what kinds of potentially incorrect assignments are present in the implementation of these four types of FMA relationships. Here we detect five major kinds of possible incorrect relationship assignments: circular, mutually exclusive, redundant, inconsistent, and missed entries. A domain expert then reviews random samples of presumptive error results to validate the algorithm used and to confirm whether these findings are in fact errors or are simply the result of a representation scheme peculiar to the model or both.

3.1. Circular relationship assignments

In order to find circular relationship assignments, we need to identify all cycles. Consider classes C_1 and C_2 and a transitive relationship r in an ontology. If there exist two links of r $C_1 \rightarrow C_2$ and r $C_2 \rightarrow C_1$, then the structure formed is called a cycle (Fig. 1a). The relationships in the cycle are circular relationship assignments. Moreover, the cycle can consist of more than two classes (Fig. 1b) and by transitivity, every class in this cycle is both the descendant and ancestor of itself. For the four types of relationships being examined here, it is obvious that such assignments are inappropriate and therefore should not exist. Once a cycle is identified, all relationships in the cycle must be checked in order to identify which assignments are acceptable and which ones have to be removed to break the erroneous circularity.

Furthermore, cycles may involve more than one type of relationship. In Fig. 1c C_1 and C_2 have two different types of relationships r_1 $C_2 \rightarrow C_1$ and r_2 $C_1 \rightarrow C_2$, e.g., r_1 is *subclass* and r_2 is *part_of*. Therefore, the cycle in Fig. 1c is formed by more than one type of transitive relationships and should not exist either. Relationships in this kind of cycle must likewise be checked.

3.2. Mutually exclusive relationship assignments

Between two classes, there may exist more than one type of relationship. However, a class should not be, at the same time, both

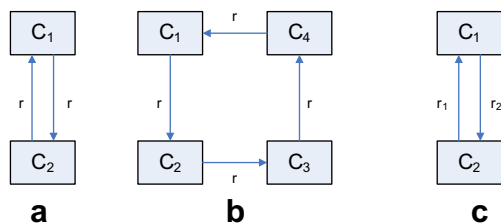


Fig. 1. Circular relationship assignments.

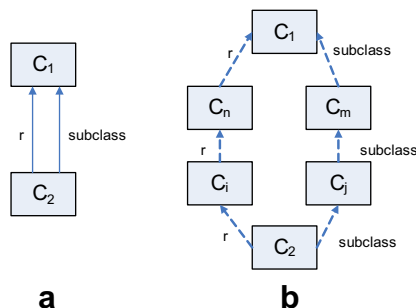


Fig. 2. Mutually exclusive relationship assignments.

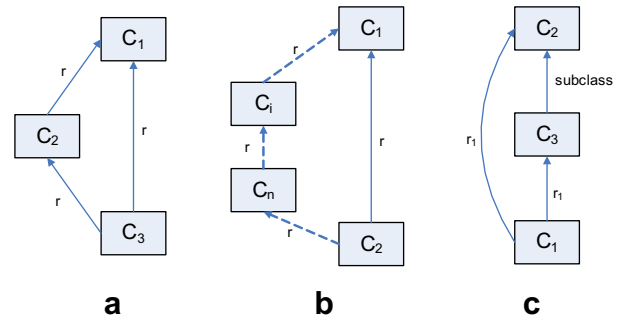


Fig. 3. Redundant relationship assignments.

a *subclass* and *participate in a part_of* (or *branch_of*, or *tributary_of*) relationship to another class. In Fig. 2a, class C_2 is a *subclass* C_1 and C_2 is *part_of* C_1 at the same time. One of these relationships must be incorrect and should be removed after a review by a domain expert. Notice that such a structure can be transitive and involve more than two classes in any single path. In Fig. 2b, there are two paths from C_2 to C_1 : one is a *subclass* path and the other is a *part_of* path.

3.3. Redundant relationship assignments

Consider classes C_1 , C_2 , and C_3 and a transitive relationship r . If there exist three links of r $C_3 \rightarrow C_2$, $C_2 \rightarrow C_1$, and $C_3 \rightarrow C_1$ (Fig. 3a), then the link $C_3 \rightarrow C_1$ is a redundant relationship assignment. Since relationships $C_3 \rightarrow C_2$ and $C_2 \rightarrow C_1$ imply $C_3 \rightarrow C_1$ by r 's transitivity, there is no need to explicitly specify the relationship $C_3 \rightarrow C_1$. Moreover, if there is a path from C_2 to C_1 , such that $C_2 \rightarrow C_n \dots C_i \rightarrow C_1$, and $C_2 \rightarrow C_1$ (Fig. 3b), then $C_2 \rightarrow C_1$ is a redundant relationship assignment and should be removed. While the previous assumption holds true for a minimal representation approach where inherited attributes and values are implicit and not declared explicitly, the FMA does not utilize a minimal approach and therefore some cases of redundant assignments for a single relationship type may exist.

Moreover, the interplay between the *subclass* and other structural relationships can also result in redundant relationship assignment. For example, there are three classes C_1 , C_2 , and C_3 and two types of relationships r_1 and r_2 in Fig. 3c. Assume r_1 is *part_of* and r_2 is *subclass*. Then C_1 is *part_of* C_2 , C_1 is *part_of* C_3 , and C_3 is *subclass* of C_2 . However, since C_3 is *subclass* of C_2 , C_3 must be a more specific class than C_2 . If C_1 is *part_of* C_3 , it is *part_of* C_2 by inference. There is no need to explicitly state C_1 is *part_of* C_2 . Therefore, C_1 *part_of* C_2 is a redundant relationship.

3.4. Inconsistent relationship assignments

Another kind of suspect relationship assignment detected by our methods involves a pattern where classes C_1 , C_2 , C_3 , and C_4 are related by two types of transitive relationships r_1 and r_2 , e.g.,

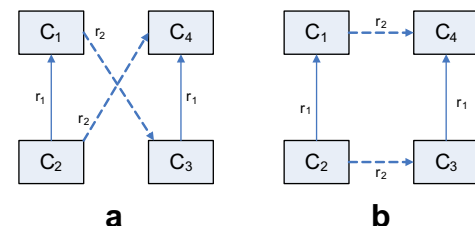


Fig. 4. (a) Inconsistent relationship assignments and (b) rectified relationship assignments.

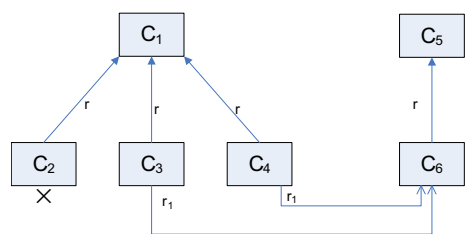
subclass and *part_of*, respectively, in the following manner: C_2 is *subclass* of C_1 and C_3 is *subclass* of C_4 , while C_1 is *part_of* C_3 and C_2 is *part_of* C_4 , as shown in Fig. 4a. Here, classes C_2 and C_3 are more specific and detailed than C_1 and C_4 but C_2 is *part_of* a more general class C_4 while its parent C_1 is *part_of* more specific and detailed class C_3 . There is clearly a conflict in the assignments. In many cases such conflicts can be rectified by assigning C_1 *part_of* C_4 and C_2 *part_of* C_3 (Fig. 4b).

The inconsistent assignments are not limited to only one level of *subclass* relationships but can appear along in the *subclass* path. For example, **Plasma membrane of enteroendocrine cell** has *part_of* **Enteroendocrine cell** while the child **Plasma membrane of Type D1 enteroendocrine cell** has *part_of* **Material anatomical entity**. However, **Enteroendocrine cell** is a several generations descendant of **Material anatomical entity**. This type of relationship assignments violates the ontology's basic principle of inheritance refinement, where a descendant slot attribute cannot be less specific than the ancestor's same slot attribute.

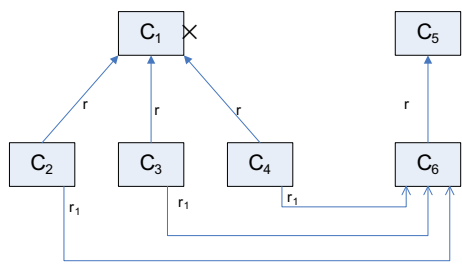
3.5. Missed relationship assignments

In the FMA, each class has one parent and may or may not have one or more children, or siblings. Siblings are classified based on shared attributes and therefore share similar semantic associations with other related classes which are either their parents or children. All **AT** classes are expected to pass down their attributes and to some extent their values to their children. On the other hand, children classes might have relationships other than those inherited from their parents. However, if all siblings share a specific relationship, it might be expected of their parent to have the same quality.

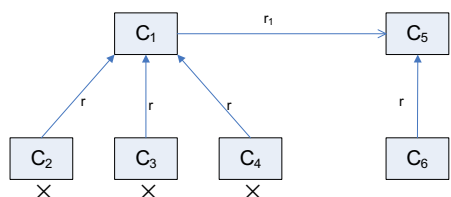
In order to test these assumptions the following three cases were checked:



(a) Sibling missing relationship assignments



(b) Parent missing relationship assignments



(c) Child missing relationship assignments

Fig. 5. Missed relationship assignments.

(a) *Sibling missing relationship*: C_2 's siblings C_3 and C_4 have the same relationship but C_2 does not have it (Fig. 5a). This may indicate that the relationship r_1 $C_2 \rightarrow C_6$ or $C_2 \rightarrow$ any descendant of C_5 is missing. The attribute value will be satisfied as long as it conforms with the definition of the slot and the facet. It does not require the attribute value to be identical for all siblings.

(b) *Parent missing relationship*: C_2 , C_3 and C_4 have the same relationship to C_6 but their parent C_1 does not carry that relationship (Fig. 5b).

(c) *Children missing relationship*: There is a relationship r_1 $C_1 \rightarrow C_5$ but none of C_1 's children C_2 , C_3 and C_4 has this relationship to C_5 or its children (Fig. 5c). This may indicate the relationships for C_2 , C_3 and C_4 are missing.

These cases may not represent true errors since due to the process of specialization concepts may acquire unique qualities that might not be shared with their parents or siblings. However, the above mechanisms are a reasonable approach for detection of omitted or neglected relationship assignments. The concepts reported by these algorithms, most likely represent lower likelihood of errors and should always be closely scrutinized by domain experts for appropriateness.

The reverse might also be indicative of potential errors. It is not always necessary or correct that a parent will have identical attributes and values as its children. However, we believe that this matter depends much more on subjective modeling preferences.

4. Results

Table 1 illustrates the number of presumptive errors in the FMA for the five categories we have examined with our auditing techniques. Overall 7052 presumptive errors were identified. An FMA domain expert's (J.L.V.M) opinion was solicited to verify whether the findings are in fact valid errors and inconsistencies. In the categories with large numbers of presumptive errors random samples were evaluated.

4.1. Circular relationship assignments

No circular errors with the same or multiple relationship types were found (Fig. 1a–c).

4.2. Mutually exclusive relationship assignments

Two hundred and sixty-six mutually exclusive relationship assignments were identified (Fig. 2), mostly related to the *part_of* relationship. All 39 samples reviewed by the FMA expert are confirmed errors. For example, the class **Set of thoracic viscera** is *subclass* and *part_of* **Set of viscera** at the same time. Another example is that class **Muscle layer of large intestine** has a grandchild **Taenia omentalis** which has *part_of* relationship to its own grandparent **Muscle layer of large intestine**. Relationships such as in the above examples violate the logical order required by the ontology and are likely an oversight committed by the authors.

4.3. Redundant relationship assignments

Out of 287 presumptive redundant relationship assignments were found with the *part_of* relationship type. For example (Fig. 3c), **Unilaminar epithelium** is a *subclass* of **Portion of epithelium** but the class **Epithelial cell** is entered as *part_of* **Unilaminar epithelium** and **Portion of epithelium**. This example most likely represents a simple case where the ontology editors lost track of the parthood assignment. **Epithelial cell** should only be *part_of* **Unilaminar epithelium**. Another example of this type of error is

Table 1
Distribution of four kinds of possible errors and analysis of samples.

Error types		Relationship types			Total number of Possible errors
		<i>Part_of</i>	<i>Branch_of</i>	<i>Tributary_of</i>	
No. of classes	With slots	78,542	13,284	3574	
	With slot values	28,492	6808	1052	
Circular relationship assignments	Total number of findings	0	0	0	0
Mutually exclusive relationship assignments	Total number of findings (findings/concepts)	244 (0.86%)	19 (0.28%)	3 (0.29%)	266
	Number of reviewed samples	17	19	3	
	Number of confirmed errors	17	19	3	
	Percent of samples with errors	100	100	100	
Redundant relationship assignments	Total number of findings (findings/concepts)	279 (0.98%)	3 (0.04%)	5 (0.48%)	287
	Number of reviewed samples	20	3	5	
	Number of confirmed errors	14	3	5	
	Percent of samples with errors	70	100	100	
Inconsistent relationship assignments	Total number of findings (findings/concepts)	7 (0.03%)	0 (0)	0 (0)	7
	Number of reviewed samples	7	0	0	
	Number of confirmed errors	7	0	0	
	Percent of samples with errors	100%	0	0	
Missed relationship assignments	Total number of sibling missing relationships (findings/concepts)	321 (1.13%)	84 (1.23%)	14 (1.33%)	419
	Number of reviewed samples	20	10	14	
	Number of confirmed errors	19	10	14	
	Percent of samples with errors	95	100	100	
	Total number of parent missing relationships (findings/concepts)	3212 (11.27%)	357 (5.24%)	108 (10.27%)	3677
	Number of reviewed samples	58	10	10	
	Number of confirmed errors	46	2	2	
	Percent of samples with errors	79	20	20	
	Total number of child missing relationships (findings/concepts)	2176 (7.63%)	174 (2.56%)	46 (4.37%)	2396
	Number of reviewed samples	60	10	10	
	Number of confirmed errors	60	10	10	
	Percent of samples with errors	100	100	100	
	Total number of possible errors		6239	637	176

the class **Set of pelvic viscera** that is *part_of* three classes **Set of viscera**, **Set of viscera of abdomen**, and **Pelvis**. Redundancy exists because **Set of viscera of abdomen** is a *subclass* of **Set of viscera**. However, **Set of pelvic viscera** is also a *subclass* of **Set of viscera**. This is a mixed example where, **Set of pelvic viscera** is both *subclass* and *part_of* **Set of viscera**. **Set of pelvic viscera** should only be a *subclass* of **Set of viscera** and a *part_of* **Set of viscera of abdomen**.

A significant portion of our findings in this category were related to the redundant assignment of *part_of* relationship of **Human body**, **Female human body**, and **Male human body** to the same concepts. Both female and male human bodies are *subclasses* of **Human body** and therefore by transitivity, it seems unnecessary to assign, for example, **Musculoskeletal system** as *part_of* **Human body** while it is also *part_of* **Female human body** and **Male human body**. However, these instantiations were created by design and will be explained in Section 5.

4.4. Inconsistent relationship assignments

Only seven instances of inconsistent relationship assignments (Fig. 4) were detected in the FMA, all for the *part_of* relationship. For example (Fig. 6), the class **Posterior radiate sternocostal ligament of seventh sternocostal joint** is a *subclass* of **Radiate sternocostal ligament of seventh sternocostal joint** and the class **Capsule of seventh sternocostal joint** is a *subclass* of **Capsule of sternocostal joint**. However, **Radiate sternocostal ligament of seventh sternocostal joint** is *part_of* **Capsule of seventh sternocostal joint** while the more specific class **Posterior radiate sternocostal ligament of seventh sternocostal joint** is *part_of* the more

general class **Capsule of sternocostal joint**. This is an example of the kind of error which can be easily made during manual data entry but readily detected by our methods.

4.5. Missed relationship assignments

By far, the bulk of presumptive errors were located in the missed relationship assignments and mostly for the *part_of* relationship type. It is essential, however, to remember that these are considered presumptive errors and, as discussed earlier, must be closely reviewed since the missed assignments captured by our algorithms in this category might be justified.

On examination, 100% and 95% of samples with presumptive errors in *child missing relationship* and *sibling missing relationship* types were confirmed as true errors by the FMA domain expert. However, 21% of presumptive errors regarding *parent missing relationship* were found to be false positives. In these cases non-instantiation of the slot may be justified (Fig. 5b). The assignment is not necessarily required at the higher, more general level, since no additional information or knowledge is gained. This is true in the case of the parent class **Subdivision of back of neck** where it is redundant to represent that some subdivision of the back of the neck is *part_of* the back of the neck. It affords no utility for any application. However, relationship assignments for the children of **Subdivision of back of neck** such as **Superior part of back of neck** and **Inferior part of back of neck** provides information of clinical import associated with the regional partition of the back of neck related in describing the location of a pathology or the spread of a tumor. Therefore our methods concentrated on the missing relationships for all three error types.

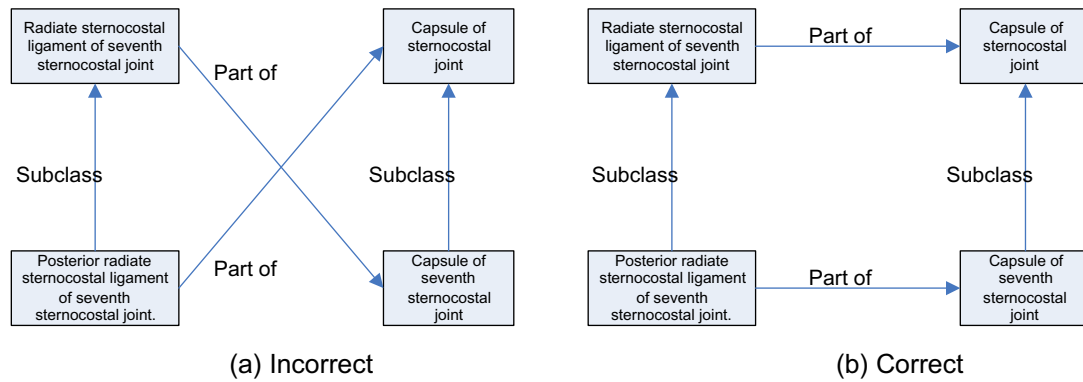


Fig. 6. Example of inconsistent relationship assignments.

As for the two other relationships in this category *sibling missing relationship* and *child missing relationship*, it might be quite likely that *sibling missing relationship* may not actually indicate an error. However, the FMA domain expert deemed all *sibling missing relationship* presumptive errors detected by our algorithm as valid errors. This high rate might indicate that although siblings may have different values for an attribute, having no value altogether while most other sibling do, is highly suggestive for an error as illustrated in the examples below.

One of *sibling missing relationship* examples (Fig. 5a) is the class **Interlobar surface of lobe of lung** that has three children **Interlobar surface of upper lobe of lung**, **Interlobar surface of middle lobe of lung**, and **Interlobar surface of lower lobe of lung**. Both **Interlobar surface of upper lobe of lung** and **Interlobar surface of lower lobe of lung** are *part_of* **Surface of lung**. However, the *part_of* relationship assignment for **Interlobar surface of middle lobe of lung** is missing. It is obviously inconsistent since that all three siblings should be *part_of* **Surface of lung**. A *part_of* relationship for **Interlobar surface of middle lobe of lung** to **Surface of lung** should be added to make it consistent. Another example of a sibling missing relationship assignment is **Intestine**. All its siblings, **Foregut**, **Midgut**, and **Hindgut**, are *part_of* **Gut** and **Gastrointestinal tract**. But **Intestine** is only *part_of* **Gut**. In this case the relationship *part_of* from **Intestine** to **Gastrointestinal tract** is missing.

As opposed to the two types above, *child missing relationship* is a clearer case of a presumptive error. Due to the governing rules of the ontology no subclass should lose attributes or attribute values, at the most a subclass attribute value can be refined. An example of *child missing relationship* (Fig. 5c) is the class **Articular circumference of head of radius** that is *part_of* **Surface of proximal epiphysis of radius**. However, the children **Articular circumference of head of right radius** and **Articular circumference of head of left radius** do not have a *part_of* relationship. In fact, the child classes should be *part_of* **Surface of proximal epiphysis of right radius** and **Surface of proximal epiphysis of left radius** respectively. Furthermore, it can be argued that the *part_of* relationship for **Articular circumference of head of radius** to **Surface of proximal epiphysis of radius** may not be necessary if its children have the appropriate *part_of* relationships.

5. Discussion and conclusions

The FMA was developed as a computable reference ontology of anatomy that can be used and reused by knowledge-based applications. Beyond the disciplined foundational principles of the FMA, the implementation relies on a frame-based representation utilizing the Protégé authoring environment. In the FMA frame-based representation, facet definitions are utilized to set the range and

type of values that can be introduced into slots. Class slots and to some defined extent, slot values are propagated down to the leaves of the hierarchy. Specialization of classes, slots, facets and slot values occurs as the hierarchical tree is traversed downwards. As such, more constraints are established along the way as they become more refined down the hierarchy.

It is to be expected that the combination of the principled design of the FMA and the rigorous authoring environment provided by Protégé should help prevent the introduction of errors within the ontology. However, this depends on the ability to enforce the rules at the time of ontology creation and during the editing process. Due to the size and complexity of the FMA, errors are likely introduced especially with manual data entry. These errors are not readily identified with simple inspection by the authors.

In this paper, we described several algorithms that we have developed to detect and uncover possible violations of the foundational principles of the FMA. The algorithms are, in a sense, re-validation of the FMA foundations and are limited to three types of structural relationships; *part_of*, *branch_of* and *tributary_of* and their interaction with the subclass relationship. Samples of findings obtained from these methods were evaluated by an FMA domain expert (J.L.V.M) to test the accuracy and efficiency of the methods used. Certain presumptive error types discovered by the automated algorithms were in fact deemed acceptable by the domain expert as in the case of the redundant relationship assignments related to the *part_of* instantiation of the human, male and female bodies and in cases involving parent missing relationship assignments (discussed below).

Our methods suggest potential errors in about 20% of FMA concepts with populated slot values in the hierarchies that were checked or 9% of all concepts. However, since certain overlap might exist between different error types per concept the true rate of presumptive erroneous concepts is likely to be somewhat lower. Our results show that presumptive errors largely involved cases dealing with *part_of* relationships, at least in terms of absolute numbers. The different rates of presumptive errors between the three relationship types might be due to the different nature and semantics of the different domains as well as the different teams of domain experts that authored these domains.

The most presumptive errors were found in the *part_of* relationship and most were related to the missed relationship assignment category. These were mostly present in the children missing relationship assignment, presumably due to the fact that the FMA editors have not yet entered the appropriate assignments. However, while this may be the reason for our large number of findings in this category, certified releases of any ontology should be complete or should at least flag incomplete areas in the ontology. It is likely that the task of completing all the assignments in the FMA will not be achieved but at least our methods can easily flag incomplete

areas in the ontology so that other applications relying on that portion of the ontology may treat the deficiencies accordingly, largely by entering the missing assignments.

There are also instances in the parthood representation where child classes have *part_of* relationships, but their parents do not, as in the case of **Right upper limb**. **Right upper limb** is *part_of* **Human body**, but the former's parent class **Limb** has no *part_of* relationship assignment to **Human body**. From the ontological point of view, a representation must approximate, as closely as possible, what exists in reality at the instance level for the individual human being [23]. Based on the FMA canonicity a person has right and left upper limbs and right and left lower limbs, not just a limb. Therefore an instance of **Human body** such as John Doe has parts **John Doe's right upper limb**, **John Doe's left upper limb**, etc. In such cases inheritance of *part_of* slot values does not apply but automated algorithms have difficulty detecting such private cases and will flag them as potential errors. In a sample of 58 presumptive errors regarding *parent missing part_of relationship* 79% were identified as true errors while for *branch_of* and *tributary of* relationship types only 20% of their respective samples were deemed relevant.

While superclasses may not always have relationship assignments that their descendants might have, the reverse case when parent classes have relationship assignments but their children are missing such assignments, is an indication that slot value instantiation has not been carried out. We see this in cases such as the class **Arteria radialis indicis** which is a *branch_of* **Radial artery** but its children **Right arteria radialis indicis** and **Left arteria radialis indicis** do not have any *branch_of* relationship. The correct relationship is that **Right arteria radialis indicis** and **Left arteria radialis indicis** have *branch_of* relationships with **Right radial artery** and **Left radial artery** respectively. As described in an earlier section, most FMA classes inherit only the relationships (Protégé slots) but not the values for those slots. Specific slot values are only applied at a particular class level and not down the hierarchy. Therefore the slot value **Radial artery** for the parent class **Arteria radialis indicis** is not propagated down as slot value to the *branch_of* slot of **Right arteria radialis indicis** and **Left arteria radialis indicis**. This principle is cardinal to our methods where missing slot values for children classes which have not yet been entered are readily identified.

A distant second source of presumptive errors comes from redundant assignments, constituting 287 errors. We encountered many instances where anatomical entities have *part_of* relationships to the three classes **Male human body**, **Female human body** and **Human body** and are consequently flagged by our method as potentially redundant. However, in the FMA there is an ontological and operational need to represent the *part_of* relationship for all three separate and independent structures, the canonical human body and the two sexually dimorphic types of the human body. It was therefore, for example, necessary for the FMA editors to enter **Musculoskeletal system** as *part_of* of all three bodies to complete the representation of different organ systems for each model. There are cases, however, where assignments are not complete, as illustrated by **Urinary system**, which is *part_of* **Human body** and **Female human body** but not **Male human body**. Such examples demonstrate the kind of inconsistencies that were most likely overlooked during manual authoring.

A striking aspect of our results is the near all-or-nothing nature of the validation by the domain expert as displayed in Table 1. For many sub-categories the sample results are either 100% or 0% validation rates. Larger samples were beyond the scope of this paper but we do assume that with larger samples validation rates may decline and vary. However, the high-validation rate in most categories demonstrates that actual errors do exist in a well-structured and audited ontology such as the FMA and that algorithmic meth-

ods do play a role in identifying such errors that were missed by audit experts due to the size and complexity of the ontology itself.

Our methods did not detect any true cyclical errors within the FMA. This might be attributed to prior work regarding such errors and the ability of the Protégé authoring environment to prevent such errors. On the other hand, the algorithms reported 266 presumptive errors related to mutually exclusive relationship assignments which were all validated to be true errors. Most of these errors are between the *subclass* and the *part_of* relationship. For example, **Wall of heart** has a child concept **Wall of right side of heart** whose *part_of* relationship assignment is to its own parent **Wall of heart**. In some respect these errors and cyclical errors are similar. While cyclical errors create an area bounded by relationships with a consistent direction, mutually exclusive relationships form an area bounded by at least two parallel mutually exclusive. Non-transitive errors of this type, should be relatively easy to detect in an authoring tool. However, we can assume that not all relationships are entered at the same time and since all the errors detected involved one or more transitive path, they may also indicate usability limitations of current design of the authoring tool where non-*subclass* relationships to one's non-immediate ancestors or descendants cannot be easily perceived.

While several of our algorithms detect only potential issues that require validation by domain experts others detect clear errors. Errors such as mutually exclusive relationship assignments, inconsistent relationship assignments or child missing relationships can be prevented algorithmically. This indicates a lack of functionality within the authoring environment, implementation of foundational principles in a way that prevents enforcement or a combination of both, thus potentially eliminating opportunities to detect many of the error types described here during the authoring process.

As well defined as the foundational principles of any ontology may be and even with sophisticated authoring tools such as Protégé, the sheer size of the ontology and its semantic complexity render any manual editing process prone to entry errors. In many cases the complexity is truly beyond comprehension. The auditing methods and their findings described in this work rely on some basic principles. These principles are not private to the FMA nor to the Protégé modeling environment and can be extended to other ontologies modeled in Protégé or other authoring environments. Our programs should require minor modifications in order to be reused with other ontologies modeled in Protégé. As for other authoring tools, our code will require significant modification or rewrites however the core logic will remain mostly the same. The methods presented by us in this paper are not intended to be comprehensive, complete or absolute. Many additional algorithms can be developed to detect ever more granular issues. The degree of certainty of presumptive errors generated by various such algorithms need not be high or definitive. The ultimate judgment as to the usefulness of the results relies with the users. Users at different roles should balance the sensitivity and specificity of the various algorithms to meet their needs. Even low yielding algorithms may detect significant errors that otherwise might have been overlooked, and with reference ontologies such as the FMA low yield might be acceptable. We do not anticipate that the error detection rate for the various methods described in this paper will be similar for other ontologies since each such ontology is subject to many objective and subjective factors. However, upon initial runs of such methods on various sized samples, we foresee that users will be able to select methods relevant for them and turn off those that they deem non-productive. This cycle can be repeated and adjusted as needed.

Not only do our findings indicate the need for consistent auditing of ontologies such as the FMA but that such principles can be automated and embedded within the ontological editing process to allow for real-time checks of attribute and relationship assign-

ments during data entry or modification of concepts and concept attributes. Running our set of programs on the complete FMA is currently a relatively time consuming process. We do expect the efficiency of our methods to improve in the future but we do not envision performing a full complement of such algorithms on a large ontology such as the FMA on a frequent basis. However, each concept in the ontology has a relatively small neighborhood of related concepts and the algorithms can be applied to the individual concept within its well-defined neighborhood. Under such circumstances response time is sufficiently low to support real-time, embedded solutions. Not only do our findings indicate that significant errors may reside within a well-structured ontology authoring tool such as Protégé, they also suggest that the visualization tools available limit the capacity of editors to view errors that are not within the immediate neighborhood of the focal concept as might be indicated by the transitive nature of the inconsistent structural relationship errors described above. While better graphical user interfaces will undoubtedly be developed in the future, combining them with real-time error detection will make them even more powerful authoring tools.

Auditing is mostly perceived as a retroactive process where errors are detected asynchronously of their introduction. Our methods and findings presented in this paper are essentially retroactive as well but offer a focused and time-effective approach to better utilize limited auditing resources. Additionally, the algorithms described in this paper are computable. Some may be fully automated while others can only generate warnings that will require expert human review. Most can be integrated into the human editing process to alert editors of potential conflicts and errors involving cyclic violations, missing, incomplete and inconsistent assignments and redundant assignments. Such integration can offer real-time auditing assistance to editors of terminologies such as the FMA.

Although the disciplined approach of the FMA resulted in a relatively low occurrence of presumptive errors, these errors may still be significant enough to affect use by particular applications. Therefore ontologies require auditing that provides quality control for building their content through assured consistency and accuracy of data entry. We propose that focused retrospective computable auditing methodologies such as ours are essential complementary tool at times when embedded algorithms are not yet available and when manual expert review is not feasible on a more consistent and permanent basis for very large ontology projects such as the FMA.

Acknowledgment

This work was partially supported by the NIH Grant HL087706.

References

- [1] Rosse C, Mejino JL, Modayur BR, Jakobovits R, Hinshaw KP, Brinkley JF. Motivation and organizational principles for anatomical knowledge representation: the Digital Anatomist Symbolic Knowledge Base. *JAMIA* 1998;5:17–40.
- [2] Rosse C, Mejino JLV. A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *J Biomed Inform* 2003;36:478–500.
- [3] Rosse C, Mejino JLV. The foundational model of anatomy ontology. In: Burger A, Davidson D, Baldock R, editors. *Anatomy ontologies for bioinformatics: principles and practice*. Springer; 2007. p. 59–117.
- [4] Min H, Perl Y, Chen Y, Halper M, Geller J, Wang Y. Auditing as part of the terminology design life cycle. *JAMIA* 2006;13(6):676–90.
- [5] Cimino JJ. Auditing the unified medical language system with semantic methods. *JAMIA* 1998;5:41–51.
- [6] Cimino JJ. Battling Scylla and Charybdis: the search for redundancy and ambiguity in the 2001 UMLS metathesaurus. In: Bakken S, editor. *Proc. 2001 AMIA annual symposium*; 2001. p. 120–4.
- [7] Gu H, Perl Y, Elhanan G, Min H, Zhang L, Peng Y. Auditing concept categorizations in the UMLS. *Artif Intell Med* 2004;31(1):29–44.
- [8] Bodenreider O. Strength in numbers: exploring redundancy in hierarchical relations across biomedical terminologies. *AMIA Annu Symp Proc* 2003:101–5.
- [9] Chen Y, Gu H, Perl Y, Geller J, Halper M. Structural group auditing of a UMLS semantic type's extent. *J Biomed Inform*; 2008 [Epub ahead of print].
- [10] Lindberg DAB, Humphreys BL, McCray AT. The unified medical language system. *Methods Inf Med* 1993;32:281.291.
- [11] Ceusters W, Smith B, Kumar A, Dhaen C. Mistakes in medical ontologies: where do they come from and how can they be detected? *Stud Health Technol Inform* 2004;102:145–63.
- [12] Ceusters W, Smith B, Kumar A, Dhaen C. Ontology-based error detection in SNOMED-CT. *Stud Health Technol Inform* 2004;107(Pt. 1):482–6.
- [13] Bodenreider O, Smith B, Kumar A, Burgun A. Investigating subsumption in SNOMED CT: an exploration into large description logic-based biomedical terminologies. *Artif Intell Med* 2007;39(3):183–95.
- [14] Wang Y, Halper M, Min H, Perl Y, Chen Y, Spackman KA. Structural methodologies for auditing SNOMED. *J Biomed Inform* 2007;40(5):561–81.
- [15] Zhang S, Bodenreider O. Investigating implicit knowledge in ontologies with application to the anatomical domain. *Pac Symp Biocomput* 2004:250–61.
- [16] Zhang S, Bodenreider O. Law and order: assessing and enforcing compliance with ontological modeling principles in the foundational model of anatomy. *Comput Biol Med* 2006;36:674–93.
- [17] Ceusters W, Smith B, Goldberg L. A terminological and ontological analysis of the NCI thesaurus. *Methods Inf Med* 2005;44(4):498–507.
- [18] Cohen B, Oren M, Min H, Perl Y, Halper M. Automated comparative auditing of NCIT genomic roles using NCBI. *J Biomed Inform*; 2008 [Epub ahead of print].
- [19] Rosse C, Shapiro LG, Brinkley JF. The digital anatomist foundational model: principles for defining and structuring its concept domain. In: *Proc AMIA Symp* 1998; 1998. p. 820–4.
- [20] Aristotle. *The categories*. Cambridge, MA: Harvard University Press; 1973.
- [21] Noy NF, Fergerson RW, Musen MA. The knowledge model of Protégé 2000: combining interoperability and flexibility. In: *Proc 12th international conference on knowledge engineering and knowledge management (EKAW-2000)*. Juan-les-Pins France: Springer; 2000.
- [22] Knublauch H, Musen Mark A, Rector Alan L. Editing description logics ontologies with the Protege OWL plugin. In: *International workshop on description logics, Whistler, BC, Canada*; 2004. Available from: <http://citeseer.ist.psu.edu/article/knublauch04editing.html>.
- [23] Grenon P, Smith B, Goldberg L. Biodynamic ontology: applying BFO in the biomedical domain. In: *Pisannelli PM, editor. Ontologies in medicine. Studies in health technology and informatics, vol. 102*. Amsterdam: IOS Press; 2004. p. 20–38.