Lifting EMMeT to OWL Getting the Most from SKOS

Bijan Parsia[1], Tahani Alsubait[1], Jared Leo[1]
, Veronique Malais[2], Sophie Forge[2], MichelleGregory[2], and Andrew Allen[2]

[1] The University of Manchester, UK
{bijan.parsia,tahani.alsubait,jared.leo}@manchester.ac.uk
[2] Elsevier B.V.
{V.Malaise,S.Forge,m.gregory,a.allen}@elsevier.com

Abstract. SKOS and OWL are quite different but complimentary languages. SKOS is targeted at "cognitive" or "navigational" representations, that is, thesauri, controlled vocabularies, and the like. OWL is targeted at logical representations of conceptual knowledge. To a first approximation, SKOS vocabularies try to capture *useful* relations between concepts whereas OWL ontologies aim to capture *true* relations between concepts. Now, of course, the true is sometimes useful and the useful often true, thus SKOS and OWL overlap to some degree. However, there are applications where we need to know true relations (e.g., generating multiple choice questions). Furthermore, SKOS relations are not precisely specified (by design). For example, many different ways of being useful can be covered by the same SKOS relation, but only one way of being useful is actually applicable to some application.
In this paper, we present a case study of modifying a large, existing SKOS vocabulary partially into OWL. This lifting is motivated by an application (generation multiple choice questions) that requires more precision in the representation than SKOS alone supports.

1 Introduction

A central use case for Web Ontology Language (OWL) Ontologies has been the development and maintenance of "terminologies" such as controlled vocabularies, taxonomies, or thesauri. Indeed, many of the most significant (in terms of longevity, funding, use, and size) ontologies such as SNOMED CT [15, 16], the NCI Thesaurus,[3] or the Gene Ontology [3] are exemplars of this use case. The use of *reasoners* to support development time services such as debugging and verification [14] as well as runtime services such as post-coordination [5] is a key part of the OWL success story for these use cases. For these, the existence of a precise formal semantics for such constructs as `SubClassOf` is a boon.

However, not all relations of interest for terminologies fit into the OWL model. Indeed, many controlled vocabularies do not need any such precision.

[3] http://ncicb.nci.nih.gov/NCICB/core/EVS

Thus, OWL is sometimes both *too strong* (e.g., `A SubClassOf B` makes an ontological commitment — that every instance of A is an instance of B — which may be the wrong one for the hierarchical relationship we want) and *too weak* (e.g., we do not have ways to indicate that terms are related *in some way or another*, at least not very easily). Hence, the introduction of the Simple Knowledge Organisation System – SKOS [10].

SKOS takes the opposite approach than OWL. Instead of insisting on a formal semantics, SKOS has an informal (or perhaps semi-formal) semantics, [4] at least for domain knowledge. SKOS is designed for representing loose *cognitive* or *navigational* relations rather than accurate domain relations. Roughly, OWL aims to model *the way the world* is whereas SKOS aims to support *how we think about it* in some context. It is easy to see how the two might come apart. Consider the mnemonic for spelling 'ocean': "Only Cats' Eyes Are Narrow". In a knowledge organisation system (KOS) about *mnemonics*, we strongly associate 'ocean' and 'cats' even though there are few domain relations between them.

Thus, SKOS allows for some flexibility and ambiguity that is a good match for many applications (such as query expansion) where the application's semantic demands are similarly loose. However, SKOS representations can become implicitly overfitted to some applications as they co-evolve. Furthermore, there are some applications where ontological representations are better suited. But migrating from a SKOS representation to an OWL on is challenging at best. As the SKOS Reference states [10]:

> To make the "knowledge" embedded in a thesaurus or classification scheme explicit in any formal sense requires that the thesaurus or classification scheme be re-engineered as a formal ontology. In other words, some person has to do the work of transforming the structure and intellectual content of a thesaurus or classification scheme into a set of formal axioms and facts. This work of transformation is both intellectually demanding and time consuming, and therefore costly...In addition, some KOS are, by design, not intended to represent a logical view of their domain. Converting such KOS to a formal logic-based representation may, in practice, involve changes which result in a representation that no longer meets the originally intended purpose.

[4] Take, for example, this paragraph from the SKOS Reference: "To understand this distinction, consider that the "knowledge" made explicit in a formal ontology is expressed as sets of axioms and facts. A thesaurus or classification scheme is of a completely different nature, and does not assert any axioms or facts. Rather, a thesaurus or classification scheme identifies and describes, through natural language and other informal means, a set of distinct ideas or meanings, which are sometimes conveniently referred to as "concepts"...These structures, however, do not have any formal semantics, and cannot be reliably interpreted as either formal axioms or facts about the world. Indeed they were never intended to be so, for they serve only to provide a convenient and intuitive map of some subject domain, which can then be used as an aid to organising and finding objects, such as documents, which are relevant to that domain."

We present a case study of ontologising a large clinical SKOS terminology, the Elsevier Merged Medical Terminology (EMMeT). EMMeT was initially released as a SKOS knowledge base. The main rationale was publish the vocabulary in a standard format for publication on the Web. Given EMMeT's non-formal structure, SKOS was a more fitting choice than OWL, in terms of standards, and was fitting for the first use cases of browsing and query expansion.

While EMMeT (as a KOS) is an excellent resource for current applications, it is not by itself suited for our application, to wit, the generation of multiple choice questions (MCQs). In particular, our MCQ generation technique requires us to distinguish between *true* and *false* subclass relations. In this paper, we describe our attempts to partially re-engineer EMMeT into an OWL Ontology.

2 Preliminaries

A *controlled vocabulary* is a collection of terms, possibly with their informal definitions. A *classification* is a controlled vocabulary that is usually, but not necessarily, hierarchically ordered. It provides a similarity-based grouping of concepts with respect to certain agreed principles, e.g., which similarity notions will be used for classifying the concepts. A *thesaurus* is a collection of concepts that can be related in three main kinds of relations: broader_than, narrower_than and related_to. The first two relations can be used to provide an informal hierarchical order of concepts while the third relation can be used to capture some notion of relevance for a given purpose, e.g., `Cars` are related_to `Fuel`. A thesaurus can also have synonymy relationships to allow for terminological level modelling. A *taxonomy* is also a collection of concepts but it is different from a thesaurus in terms of the underlying relations. In particular, a taxonomy is built using the so-called is_a relation which can provide a real subsumption hierarchy.

SKOS [10] is a World Wide Web Consortium (W3C) recommendation since 2009. It provides a lightweight language for representing knowledge in controlled structured vocabularies, classifications or thesauri and can be encoded using any concrete RDF syntax, e.g., RDF/XML. SKOS concepts can be linked to other SKOS concepts using hierarchical (e.g., `skos:narrower` and `skos:broader`) or associative relations (e.g., `skos:related`). In SKOS, the relation `rdf:type` which is used to specify instances of concepts is not available. Thus, extensional connections between concept are not modelled. For example, consider a SKOS concept about "computers" which can be associated with the concept "printer" via a broader_than relation. Clearly, not all instances of printers are also instances of computers; hence, this is not a valid subsumption relation. However, the concept "computers", in this context, may be interpreted as computers and related devices from a sales and marketing perspective. As a result, SKOS hierarchical relations are not transitive by default. For example, a "printer" might be related to, indeed broader than, "A4 paper" but, "A4 papers" are not necessarily narrower than "computers" (via "printer" being narrower than "computers") because while "A4 paper" might well be a natural more specific search from "printer", it might not be a reasonable next level search from "computers": People looking at printers often want to buy paper. Fewer want to shop for paper while considering computers. Navigationally, there is a chain but we don't want shortcuts through that chain.

The Web Ontology Language (OWL) is the W3C standard ontology language for the web and was standardised in 2004. It provides a formal knowledge representation language with unambiguous semantics, i.e., context-independent meaning. An OWL ontology is a finite set of axioms that describe the main notions, i.e., concepts, of a domain of interest. The inferred class hierarchy is the Hasse diagram of the partial order on concept names, e.g., A and B, in an ontology $\mathcal{O}$ induced by the entailment relation $\mathcal{O} \models A \sqsubseteq B$. The main relation in inferred class hierarchies is the is_a relation which is a transitive subsumption relation. Clearly, this is different from `skos:narrower` and `skos:broader` relations which are not necessarily valid subsumption relations. OWL exploits Description Logics (DLs) [1] to provide ontologies with formal semantics.

Part of OWL's success is the availability of a number of optimised reasoners such as FacT++ [17], Pellet [14], HermiT [13], and ELK [7].[5] Different ontology editing and processing tools and libraries are readily available as well such as *Protégé*,[6] and the OWL API [2].

In addition to the standard reasoning services provided by the above reasoners, some useful non-standard reasoning services have also been developed. For example, many techniques have been developed to extract modules, i.e., subsets of the axioms in a given ontology that are "relevant" to a particular signature. An interesting property of modules is that they preserve all entailments relevant to the intended signature, yet they are much smaller than the original ontology. For example, if $\mathcal{O} \models C \sqsubseteq E$, where E is a concept expressible in a considered DL, then extracting a module $\mathcal{M}$ with a seed signature $\Sigma = \{C\}$ also guarantees that $\mathcal{M} \models C \sqsubseteq E$.

The Elsevier Merged Medical Taxonomy (EMMeT) is currently modelled in SKOS. It contains 927,827 concepts with 3,010,262 synonyms in the EMMeT 3.8 release (May 15 2015). Some prominent such areas include: Anatomy (17,000 concepts), clinical findings (8,500), drugs (40,500), organisms (34,000), procedures (61,000), along with symptoms (38,000). Further more there are 132,000 semantic relationships between these concepts.

EMMeT uses 3 types of elements in their SKOS representation: `skos` and `skosxl` elements, custom nodes used to represent semantic relations, and meta data nodes. Amongst the `skos` and `skoxl` terms are elements to classify concepts, e.g., `skos:Concept`, `skos:ConceptScheme`, `skosxl:prefLabels`, and elements used to act as relations between concepts such as `skos:narrower` and `skos:broader` and `skos:ExactMatch` which expresses a relationship between concepts from EMMeT and external conceptSchemes or vocabularies. Whenever possible, EMMeT make use of existing standard properties, like the Dublin Core set for metadata, the PROV vocabulary for Provenance, RDF, SKOS and SKOS-XL. Whenever a custom property is needed, like explicit se-

[5] for a list of DL reasoners: http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/

[6] http://protege.stanford.edu

mantic relationships (which are more precise than `skos:related`), the idea is to create them as sub-properties of standard W3C properties, to keep the compatibility with other published vocabularies. The metamodel, however, was not published together with the EMMeT release as it should have been.

The namespace `semrel` (semantic relation) was used in order to be able represent a concept to concept relation and specify a *ranking* of importance the concept to concept relation has in the general knowledge base. For example,

```
<semrel:isACauseFor rdf:ID="Relation-2996187-i" .../>
```

defines a relationship between two concepts, with an ID that allows for the reification of this relationship. The reification method is used to assign a rank to that relationship. For example:

```
<semrel:Relation rdf:about="...">
    <semrel:rank>9.0</semrel:rank>
</semrel:Relation>
```

These ranks are used in several ways including to filter or order results. For example, a very low ranked related concept might only be displayed if no more high ranked related concepts are found.

EMMeT local (`emloc`) is another defined namespace which represents a very specific semantic relationship between a coordinated concept (e.g., disease due to X symptoms) and its compounds: "Disease due to X" and "Symptoms". This relationship was designed to be used in a very specific case of knowledge intensive query expansion: to link a disease to its symptoms, treatments etc and allow for the actual symptom/treatment concepts to be added to the query expansion.

```
<emloc:hasLocalChildren rdf:ID="Relation-3041760-h" .../>
<!--  Disease due to Deltaretrovirus Symptoms  -->
```

There are also other metadata properties that are used as provenance and quality assurance for concepts creation and maintenance, for example information representing creation dates (`pav:createdOn`) or versions (`sat:version`).

To illustrate the use of these nodes, consider Figure 1. This example shows an extraction of EMMeT highlighting the usage of the elements described above. The example represents a graph between the 5 following concepts:

- `breast cancer`
- `malignant melanoma of skin of breast`
- `oncology`
- `radiation therapy`

4 narrower/broader relations:

- `<breast cancer> skos:broader <malignant melanoma of breast>`
- `<malignant melanoma of breast> skos:narrower <breast cancer>`

- <malignant melanoma of breast> skos:broader <malignant melanoma of skin of breast>
- <malignant melanoma of skin of breast> skos:narrower <melignant melanoma of breast>

and 2 semantic relations:

- (malignant melanoma of breast semrel:hasPhysicianSpeciality <oncology>) rank:6.0
- (malignant melanoma of breast semrel:hasTreatmentProcedure <radiation therapy>) rank:6.0

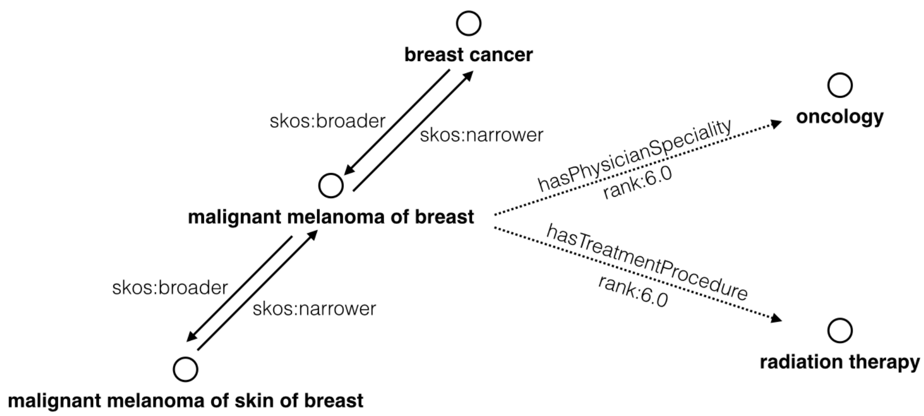

Fig. 1. An extraction from EMMeT, illustrating the use of concepts and their relations

3 Lifting EMMeT to OWL

EMMeT is clearly very large and has coverage on the scale of other major clinical terminologies. It is not feasible to do a hand crafted lifting based on term by term view by clinically savvy medical experts. We needed at least a "good enough" initial translation that hopefully will be sufficient for assessing whether EMMeT is in principle adequate for our task. To this end, we first attempted what seemed to be a straightforward if a bit naive approach on the theory that EMMeT might be close enough to OWL and if not such a translation would flush out problems. The latter proved to be the case, so we performed a more sophisticated but still automatic translation making use of the fact that many EMMeT concepts are mapped to SNOMED.

3.1 The Naive Approach

The basic idea of the naive approach is to presume that EMMeT's use of SKOS is more or less as a kind of syntax for OWL.

Classes: Clearly, on nearly any translation, we will map skos:Concept to owl:Class. In general, owl:Classes are intended to represent concepts and act as a way to describe part of a domain. Since skos:Concept's are used to describe a particular part of a knowledge base, or a term which covers over terms of similar properties, the mapping to owl:Class seemed obvious.

Relations: EMMeTs semrel:X relations are associative relations intended to relate concepts via a property. Using the example from Section 1, (malignant melanoma of breast semrel:hasTreatmentProcedure <radiation therapy>) rank:6.0 is intended to represent that a *malignant melanoma of breast* has a *treatment procedure* that is *radiation therapy*. skos:semanticRelation's are described in the documentation as equivalent to owl:objectProperty's. Since both radiation therapy and malignant melanoma of breast are classes, this leads to the following OWL modelling choice: <malignant melanoma of breast> ⊑ ∃<hasTreatmentProcedure>.<radiation therapy> where hasTreatmentProcedure is an instance of owl:objectProperty. In OWL, at an axiom level there are two ways to model property relations between two concepts. The first being universal (∀) restrictions, and the second, as used in the example, being existential (∃) restrictions. To illustrate the basic use of each example, consider the following two axioms

$$A \sqsubseteq \exists R.D \quad (1)$$
$$A \sqsubseteq \forall R.D \quad (2)$$

(1) says that every instance of A has an R-successor to an instance of D, whilst the second tells us that every instance of A must only have R successors to instances of Ds. Since we dont want to enforce the restriction that <malignant melanoma of breast> only has rediation therapy as a treatment procedure and no other treatment (EMMeT tells us that simple mastectomy, chemotherapy and mastectomy are also treatment procedures with ranks 5.0, 6.0, 6.0 respectively), we opted for the existential axiom, leaving the door open for other treatments being allowed. The same modelling options were EMMeTs relations emloc:hasLocalChildren and its inverse isLocalChildOf. Although these may be interpreted as similar to narrower and broader relations, after closer inspection this was not so the case, therefore owl:objectProperty's were used, in the same way as above. In general, existential readings are preferred in biomedical ontologies not least because it opens the possibly of staying inside the polynomial OWL EL profile, which is the profile of choice for SNOMED, among others.

Narrower and Broader: skos:narrower and skos:broader are hierarchical relations used to indicate that one concept is in some way more general (*broader*) than the another concept (*narrower*). There is really no other reasonable choice in OWL if we want to model hierarchical concept relations, at least, without a complex scheme that is unlikely to be understood by other modellers. Of course, we know that `SubClassOf` is transitive whereas the source SKOS relations are not. However, it might be that they *are* transitive when read ontologically. After all, it is a standard to use the transitive reduct of an OWL class hierarchy as a navigational structure — this is exactly what the Protg class hierarchy view does.

Ranks In EMMeT, semrel:rank properties were used to map associative relations between concepts to a rank (a measure) according to specific metrics such as pertinence to a specialty, quality of test, or severity of consequence. Given that reification is used in the SKOS model, we carried that over to our first OWL model. Returning to our example, we know that the ranking between `(malignant melanoma of breast semrel:hasTreatmentProcedure <radiation therapy>) rank:6.0`, has a value of 6.0. We model this in `owl` by introducing a new object property *hasRankedRelation* and the data property *hasRank* as follows:

$$\texttt{<malignant melanoma of breast>} \sqsubseteq \exists\texttt{<hasRankedRelation>}.(\exists\texttt{<hasTreatmentProcedure>}.\texttt{<radiation therapy>} \sqcap \exists\texttt{<hasRank>}.\texttt{<6.0>}) \quad (3)$$

This reads as every instance of the class *malignant melanoma of breast* has a *ranked relation* to an instance that both *has a treatment procedure that is radio therapy* and that *has rank 6.0*.

Meta data All meta data, including alternate labels, preferred labels, and mappings to other vocabularies or ontologies were included as non-logical assertions, namely `owl:annotation` assertions, and also using `rdfs:labels` where appropriate.

All of these transformations are performed by a simple program written in Java and using the OWL API.

Results The OWL lift resulted in an ontology with 927,941 classes with over 1.6 million asserted logical axioms. There were 45 object properties and one data property. The number of inferred atomic subsumptions stood at over 21 million. There were several issues that were revealed. The first issue was that there were numerous modelling errors from mapping narrower and broader relations to sub and superclass relations. For example, consider the classes `Abortion` and `Abortion Recovery`. It is clear that `Abortion` is a broader term than `Abortion Recovery`, hence the use of a `skos:broader` relation in EMMeT. However, to state that one is a subclass of the other is just wrong: abortion recovery is not a kind of abortion. Thus we are at high risk of generating *incorrect* keys. That is, our technique could easily generate a multiple choice question where "abortion recovery" would be treated as a correct answer when it obviously is not.

The second issue was in relation to inheritance of superclass properties. Consider the disorder `Acute left ventricular failure`. According to EMMeT, this disorder is related via `has diagnostic procedure` to the class `echocardiography`. The relation has a rank of `6.0`. A broader term for `Acute left ventricular failure` has the same procedure relation to `echocardiography` but with a rank of `10.0`, which will be inherited into `Acute left ventricular failure` due to a subclass relation; i.e. it will have two rankings. This is clearly incorrect.

3.2 A More Sophisticated Approach

Of these two issues, the first is more challenging. The ranking issue can be handled simply by using an alternative modelling construct that avoids the problematic inheritance. We do not need to *interpret* or *validate* any of the specific rankings, thus the fixing remains domain independent. This is not true for the broader-narrower relations. There, we need to know *which ones* actually represent subsumptions (as many of them obviously do).

Ranking The key to the second issues is that ranks are actually a sort of extra logical feature of EMMeT. Such extra-logical features are standardly represented with annotations and left for downstream processors to handle. OWL 2 added the ability to annotate whole axioms so this seemed like a good fit. Using the same example from Section 3.(1), we now introduce and annotation property `hasRank` in the following way

$$<(< \mathit{malignantmelanomaofbreast} > \sqsubseteq \exists\texttt{<hasTreatmentProcedure>.<radiation therapy>}) >: \texttt{<hasRank> <6.0>}) \qquad (4)$$

Annotation properties are not inherited to subclasses, and although we may lose any logical inferences w.r.t the data property version, since the annotation is basic, simple processing of the ontology would allow us to easily get the same information back.

SNOMED Alignment To address the first issue, we needed a source of domain knowledge. Manually inspection was not a feasible option due to the size of EMMeT and lack of available domain expertise. However, EMMeT does have semantic relations which associated EMMeT concepts to concepts external ontologies. Critically, one especially well connected source is SNOMED-CT[**?**]. SNOMED-CT, the coding controlled vocabulary is backed by a richly axiomitised OWL ontology and a long held focus on modelling domain relations correctly (esp. in an US context, though there are various internationalised versions and extensions). That plus the fact that it has extensive coverage of the clinical domain makes it an ideal source of "ground truth" for subsumptions.

We did first try our question generation technique using SNOMED-CT alone but found that it lacked many relations between concepts that were essential for the sorts of questions we need to generate. For example, `isClinicalFindingFor`, `hasTreatmentProcedure`, and `hasDiagnosticProcedure` are critical for formulating diagnostic puzzles, and are prevalent in EMMeT but not in SNOMED

Out of the 927,941 EMMeT-in-OWL concepts, 106,435 contained mapping relationships to equivalent SNOMED CT classes. We first decided to test the accuracy of narrower broader relations of classes with SNOMED IDs against the same classes in SNOMED to see if a subclass relation was also present. There were over 1.4 million inferred atomic subsumptions in EMMeT for which both subclass and superclass had SNOMED IDs (6% of all atomic subsumptions in EMMeT). From that 1.4 million, over 1.08 million (75%) occurred in SNOMED, leaving 355,880 (25%) not present in SNOMED. While it would still be a challenging task to review that remaining 25% (which are potentially a source of clinical knowledge beyond SNOMED), having the large number of SNOMED valid subsumption is more than sufficient for current purposes. Finally, there were 487,382 atomic subsumptions that were not present in EMMeT and were present in SNOMED. It seems somewhat implausible that these are "intentional" misses, that is, that the EMMeT designers think that SNOMED got some subsumptions wrong. These are a potential source for enriching EMMeT and we are investigating that possibility.

We identified two methods of including SNOMED CT in EMMeT: 1. Importing SNOMED directly and aligning the equivalent concepts and 2. Add the subclass relations to existing EMMeT concepts. We also opted to encode`skos:narrower` and `skos:broader` relations as owl object properties instead of subclass and superclass relations, keeping there original intended semantics in both alignments. We decided to implement both methods and evaluated them against each other by comparing additional entailments when computing the class hierarchy in each ontology. Surprisingly, other than the entailments SNOMED alone provides, both ontologies did not provide any additional entailments upon classifying. After closer inspection we found that this was due to how we modelled the axioms. All logical axioms in the ontology are either of the form $A \sqsubseteq B$ or $A \sqsubseteq \exists R.B$. The left hand side of each axiom is always a named class, and since we have no definitions (axioms using $\equiv$ instead of $\sqsubseteq$) no further entailments could be inferred. Unfortunately, simply strengthening the axioms to equivalences would lead to a number of bogus results. (E.g., different diseases may have a common cause, so if we made each equivalent to having that cause we would end up with distinct diseases being equivalent.) We are currently investigating alternatives.

4 Related Work

Hepp and de Bruijn (2007) [6] introduced the GenTax algorithm to transform a thesaurus into a "useful" ontology by creating two OWL classes for each concept in the original hierarchy: (1) a *Gen*eric class and (2) a *Tax*onomy class that can be used to build a subsumption-based hierarchy that is valid with respect to a particular context of interest. Human intervention is required to determine (1) the main notions of the intended context, (2) a preliminary classification that is valid in the intended context. The obvious drawback of this methodology is the increased size of the resulting ontology since each concept in the original hierarchy is transformed into two separate concepts. The SKOS2OWL[7] tool is an implementation of the GenTax algorithm which offers a script-based transformation of SKOS informal hierarchies to OWL ontologies (with limited human intervention).

[7] http://www.heppnetz.de/projects/skos2owl/

The method introduced by Wielinga et al. [19] is one of the earliest non-naive[8] transformations of thesauri to ontologies. The method was extended in [18] and [20] to transform two thesauri, namely MeSH [12] and WordNet [11], to OWL ontologies.

The work presented in this paper is also related to ontology learning techniques in that the ultimate goal of such techniques is to build an ontology from existing knowledge sources. Our approach makes use of existing thesauri, in particular SKOS knowledge bases, to build the ontology. Such knowledge bases are valuable sources for ontology learning as they already contain some hierarchical ordering between concepts and they can be rich in semantic relations. A review of other existing ontology learning techniques is out of the scope of this paper. The interested reader is referred to [4].

5 Conclusion

Given a bit of luck, we were able to produce a prototype version of EMMeT with sufficiently reliable subsumption relations for at least experimental work and perhaps even production quality generation of multiple choice questions. Futhermore, we seem to have identified an "easy win" extension of EMMeT with additional SNOMED relations. The effort involved was but a few person weeks and did not require expensive domain expertise.

On the downside, many broader-narrower relations are still "mysterious" from an ontological perspective. We just do not have enough information to know whether any of the residual relations are subsumptions (or definite *non*-subsumptions. We feel this exposes a weakness in the SKOS relaxed approach to relation specification: It inhibits reuse. Essentially, *only* applications that work at the same level of underspecification (or weaker) can use EMMeT without extensive examination. Whereas, if all the relations were *more specified* (not necessarily *as subsumptions* but at least as subsumptions *when they were* subsumptions), then use across various applications would be straight

We will continue to strengthen EMMeTs hierarchy. As well as SNOMED-CT, there are also symbolic links to other formalisms such as UMLS and ICD which can be used to provide more entailments between concepts. It will be interesting to know how many of the narrower/broader relations are present as subclass relations w.r.t these formalisms. We also are exploring OWL modelling that would safely exploit the semantic relations we are currently modelling as subsumptions of existential restrictions.

References

1. F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. (eds.) Patel-Schneider. *The Description Logic Handbook: Theory, Implementation and Applications.* Cambridge University Press, second edition, 2007.

[8] By naive, here, we mean simply converting the hierarchy in the thesaurus to a subsumption hierarchy, ignoring any possible invalid consequences, as has been done in [9] and [8]

2. S. Bechhofer, R. Volz, and P. Lord. Cooking the semantic web with the OWL API. In *ISWC-03*, 2003.
3. T. G. O. Consortium. Gene ontology: tool for the unification of biology. *Nat. Genet. May 2000*, 25(1):25–9, 2000.
4. L. Drumond and R. Girardi. A survey of ontology learning procedures. In *3rd Workshop on Ontologies and their Applications, volume 427 of CEUR Workshop Proceedings*, 2008.
5. C. Hedeler, B. Parsia, and S. Brandt. Estimating and analysing coordination in medical terminologies. In *Computer-Based Medical Systems (CBMS), 2014 IEEE 27th International Symposium on*, pages 357–362, 2014.
6. M. Hepp and J. de Bruijn. GenTax: A generic methodology for deriving OWL and RDF-S ontologies from hierarchical classifications, thesauri, and inconsistent taxonomies. In *Proceedings of the 4th European Semantic Web Conference (ESWC 2007)*, volume 4519, pages 129–144, Innsbruck, Austria, June 2007. Springer LNCS.
7. Y. Kazakov, M. Krötzsch, and F. Simancik. Unchain my EL reasoner. In *Proceedings of the 24th International Workshop on Description Logics (DL-11)*, 2011.
8. M. Klein. DAML+OIL and RDF Schema representation of UNSPSC. available at http://www.cs.vu.nl/ mcaklein/unspsc/, retrieved August 4, 2015.
9. D. L. McGuinness. UNSPSC Ontology in DAML+OIL. available at http://www.ksl.stanford.edu/projects/DAML/UNSPSC.daml, retrieved August 4, 2015.
10. A. Miles and S. Bechhofer. SKOS simple knowledge organization system reference. available at http://www.w3.org/TR/skos-reference/, retrieved August 4, 2015.
11. G. MILLER. WordNet: A lexical database for English. *COMMUNICATIONS OF THE ACM*, 38(11), November 1995.
12. S. J. Nelson. Medical terminologies that work: The example of MeSH. In *Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN 2009)*, pages 380–384, Kaohsiung, Taiwan, December 2009.
13. R. Shearer, B. Motik, and I. Horrocks. HermiT: A highly-efficient OWL reasoner. In *Proceedings of the 5th International Workshop on OWL: Experiences and Directions (OWLED-08EU)*, 2008.
14. E. Sirin, B. Parsia, B. Cuenca Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics*, 5(2), 2007.
15. K. Spackman and K. Campbell. Snomed ct: A reference terminology for health care. In *Daniel R. Masys, editor, Proceedings of AMIA Annual Fall Symposium, Bethesda, Maryland, USA*, pages 640–644. Hanley and Belfus Inc., 1997.
16. M.Q. Stearns, C. Price, K.A. Spackman, and A.Y. Wang. Snomed clinical terms: Overview of the development process and project status. In *Proceedings of the 2001 AMIA Annual Symposium, pp. 662-666. Hanley and Belfus*, 2001.
17. D. Tsarkov and I. Horrocks. FaCT++ description logic reasoner: System description. In *Proceedings of the 3rd International Joint Conference on Automated Reasoning (IJCAR)*, 2006.
18. M. van Assem, M. R. Menken, G. Schreiber, J. Wielemaker, and B. J. Wielinga. A method for converting thesauri to RDF/OWL. In *Proceedings of the ISWC'04*, Hiroshima, Japan, 2004.
19. B. J. Wielinga, A. T. Schreiber, and J. A. C. Sandberg. From thesaurus to ontology. In *Proceedings of the First International Conference on Knowledge Capture (K-CAP 2001)*, Victoria, British Columbia, Canada, 2001.
20. B. J. Wielinga, J. Wielemaker, G. Schreiber, and M. van Assem. Methods for porting resources to the semantic web. In *Proceedings of the First European Semantic Web Symposium (ESWS'04)*, Heraklion, Greece, 2004.