Representing and sharing knowledge using SNOMED
Proceedings of the 3rd international conference on Knowledge Representation in Medicine (KR-MED 2008)
R. Cornet, K.A. Spackman (Eds)

Exploiting Fast Classification of SNOMED CT for Query and Integration of Health Data

Michael J. Lawley
Queensland University of Technology, Faculty of Information Technology, Brisbane, (Queensland), Australia
E-Health Research Centre, CSIRO ICT Centre, (Queensland), Australia

Abstract

By constructing local extensions to SNOMED *we aim to enrich existing medical and related data stores, simplify the expression of complex queries, and establish a foundation for semantic integration of data from multiple sources.*

Specifically, a local extension can be constructed from the controlled vocabulary(ies) used in the medical data. In combination with SNOMED*, this local extension makes explicit the implicit semantics of the terms in the controlled vocabulary. By using* SNOMED *as a base ontology we can exploit the existing knowledge encoded in it and simplify the task of reifying the implicit semantics of the controlled vocabulary. Queries can now be formulated using the relationships encoded in the extended* SNOMED *rather than embedding them ad-hoc into the query itself. Additionally,* SNOMED *can then act as a common point of integration, providing a shared set of concepts for querying across multiple data sets.*

Key to practical construction of a local extension to SNOMED *is appropriate tool support including the ability to compute subsumption relationships very quickly. Our implementation of the polynomial algorithm for $\mathcal{EL}+$ in Java is able to classify* SNOMED *in under 1 minute.*

INTRODUCTION

Experience with integrating medical and related data [1] shows that the use of controlled vocabularies successfully modulates the amount of noise in the data. However, when querying the collected data, any semantic relationships between the terms that are relevant to the query (for example, specialisation/generalisation or part-of relationships) need to be explicitly encoded in the query and/or accounted for in the interpretation of the query results.

These kinds of implicit relationships are especially common in the health domain where terms often involve an implicit context of usage (e.g., *lobe* in the context of lung cancer) or implicit references to anatomical structures (e.g., colorectal cancer) or related classes of diseases, injuries, or procedures. Accurately and consistently encoding these relationships in queries relies on the person formulating the queries to understand them, thus creating many opportunities for errors, omissions, and inconsistencies to occur. When multiple people are constructing queries these risks are further exacerbated.

By constructing the vocabularies so as to explicitly represent the relationships between terms, queries can directly and consistently exploit the relationships. Using an ad-hoc explicit representation of these relationships helps, but may introduce new problems in terms of consistency of usage and how the relationships are interpreted (see, for example, the Radiological Electronic Atlas of Malformation Syndromes and Skeletal Dysplasias (REAMS) [2]). Instead, using a well-understood formal mechanism for representing the relationships, such as Description Logic, can avoid these problems.

However we still have two problems to solve:

1. how do we deal with all the existing data sets that do not do this; and
2. how do we mitigate the, potentially quite high[1], cost of explicitly representing all the relationships?

We can deal with both these problems by extending (as needed) an existing standard ontology, such as the Systematized Nomenclature of Medicine (SNOMED) [3], that already embodies

[1] Getting the modelling right, from scratch, requires not only an excellent understanding of the concepts involved as well as their relationships, but also an understanding of how best to represent them in a particular Description Logic formalism.

many of the relationships we need. However, one of the main difficulties with this approach is that building an extension to SNOMED is not dissimilar to maintaining and developing SNOMED itself. That is, the sheer size of SNOMED has meant that, until recently, very few tools could compute all of its subsumption relationships, and even those that could would reportedly take several hours.

Fortunately, recent work by Baader et al. [4, 5] on the tractable family of description logics $\mathcal{EL}$ has shown that polynomial time classification algorithms exist and are practical. Moreover despite their relatively low expressive power, the $\mathcal{EL}$ family of description logics is suitable for representing such real-world ontologies as SNOMED and offer additional expressiveness suitable for properly representing partOf relationships and sufficient conditions.[2] Their implementation of this algorithm in Lisp is able to classify SNOMED in 1,782 seconds [5] (approx. 30 minutes) which suggests an optimised implementation in a lower-level language may be fast enough for near real-time feedback in an editing tool.

Thus, our goal is to provide tool support for defining a *local extension* to an existing standard formal ontology; a mapping from an existing set of terms that characterise an informal ontology to concepts in the formal ontology. In doing so we effectively realise latent semantics in the existing medical data via the standard ontology. This should facilitate simpler and more robust queries and in turn aid data integration, a special-case application of querying where related medical data sets use semantically overlapping, but distinct term sets.

RELATED WORK

There is a great deal of published work on using ontologies for data integration (see Wache et al. [6] for an overview), but it is mostly focussed on their use at the meta-data level; ontologies are used to describe, reason about and integrate database schemas. While related to our goals, we are addressing the more specific problem of semantic data integration or semantic translation. Stuckenschmidt et al. [7] discuss an approach to this problem in the context of their Ontology Interchange Language (OIL) [8]. In particular they raise the question of whether it is feasible to find or create a sufficient shared terminology. In our domain of medical data we believe that SNOMED represents such a shared terminology. A possibly more important problem, and one identified in our work with skeletal dysplasias [2], is how to cope with errors in the shared terminology.

Wade and Rosenbloom [9] report on the manual construction of what is almost a local extension to SNOMED (they conceived the task as a semi-formal mapping). In this work 2002 terms were mapped to combination of single and post-coordinated concepts of which about 75% were equivalencies (20% of these were to single concepts) and only 1% (26) were, in their words, *"not mappable"*. It is unclear why these terms were categorized as such since they include, for example, *presyncope* which could reasonably be related to 3006004|disturbance of consciousness|, but it may be that the context of use of the terms was unavailable in order to properly discern their meaning. However, their work does demonstrate that the goal of producing a local extension to SNOMED is feasible.

PROBLEM DESCRIPTION

The problem of embedding domain semantics such as specialisation/generalisation or part-of relationships into queries is illustrated in the following. For example, a query to find all performed procedures involving a colectomy might enumerate all such procedures:

```
SELECT S.*
FROM Surgery S
WHERE S.procedure = '32003-00'
   OR S.procedure = '32003-01'
   OR S.procedure = '32012-00'
   ...
```

which has the potential to accidently omit certain codes and will require updating if the terminology is updated with additional forms of colectomy.

Alternatively, some kind of heuristic query could be used:

```
SELECT S.*
FROM Surgery S, ProcedureCodes C
WHERE S.procedure = C.code
  AND C.text LIKE '%colectomy%';
```

which has the potential to miss a term that doesn't follow the expected naming pattern (e.g., epiploectomy) or provide false matches where a compound or composite name does not reflect a valid specialisation.

If, however, the terms were encoded as concepts in an ontology, the query is simple[3]:

[2] See also `http://webont.org/owl/1.1/tractable.html#2`

[3] We envisage that the complete set of subsumption relationships would be stored in a database table to support fast subsumption-based queries using only two

```
SELECT S.*
FROM Surgery S, Ontology O
WHERE O.ancestor = 23968004
  AND S.procedure = O.descendant;
```

Note also that SNOMED, unlike classification schemes such as ICD-9 and ICD-10, support a multi-parented generalisation hierarchy.

CONSTRUCTING LOCAL EXTENSIONS

In order to construct an ontology from an existing terminology (or collection of terminologies) we take a multi-step approach:

1. Map each term from the controlled vocabulary to a concept, factoring out any synonyms, to produce $\mathcal{P}$.

   This is often a simple one-to-one mapping, but it may be necessary to extend the mapping to include disambiguating data values when the same term is used to mean different things in different contexts.

2. Make any simple implicit relationships explicit, adding them to $\mathcal{P}$.

   For example, generalisation, *partOf*, or *hasLocation* relationships. It may be necessary to introduce new concepts to act as the generalisation of two or more sibling concepts.

3. Specify relationships between these (local) concepts and those in the chosen standard ontology $\mathcal{Q}$, adding them to $\mathcal{P}$.

To be able to answer queries involving our new ontology we first need to classify $\mathcal{Q} \cup \mathcal{P}$ to identify all the subsumption relationships it entails.
Note that, we should be careful that $\mathcal{Q} \cup \mathcal{P}$ represents a conservative extension [10] of $\mathcal{Q}$. That is, $\mathcal{Q} \cup \mathcal{P}$ produces the same consequences over the set of concepts in $\mathcal{Q}$ as $\mathcal{Q}$ does by itself. We also need to ensure various integrity constraints (such as disjointness) are preserved in $\mathcal{Q} \cup \mathcal{P}$. Thus we would like to be able to interactively edit $\mathcal{P}$ while exploiting the consequences of $\mathcal{Q} \cup \mathcal{P}$ in live feedback through the mapping tool. These kinds of checks can be performed by classification of $\mathcal{Q} \cup \mathcal{P}$ but this may not be viable if $\mathcal{Q} \cup \mathcal{P}$ is large, as is the case when $\mathcal{Q}$ is SNOMED.

Colorectal Cancer Example

In this section we consider a sample set of ICD-10-AM [11] terms for procedures relating to colorectal cancer, shown in Figure 1. We can map these, one-to-one, to a set of concepts for a local ontology.

Procedure Code (ICD-10-AM)	Meaning
32000-00	Sig colectomy *with stoma formation*
32003-00	Sig colectomy *with anastomosis*
32003-01	Right hemicolectomy
32005-00	Subtotal colectomy
32005-01	Ext right hemicolectomy
32006-00	Left hemicolectomy
32012-00	Total colectomy
32024-00	High anterior resection
32025-00	Low anterior resection *extraperitoneal*
32026-00	Low anterior resection *coloanal anastomosis*
32028-00	Ultra low anterior resection
32030-00	Hartmann's procedure
32039-00	Abdomino-perineal excision
32051-00	Total proctocolectomy with ileo-anal anastomosis

Figure 1: A Term-Set of Colorectal Cancer Procedures

The next step is to make any simple relationship explicit. In our case there are none that can be expressed using just the concepts we have currently identified.

Figure 2 describes the identified relationships between these terms and selected SNOMED concepts as per step 3. Note that several concepts (for example, 32028-00|ultra low anterior resection|), have no exact equivalent in SNOMED, and that one, 32051|total proctocolectomy with ileo-anal anastomosis| implies a composite of concepts.

Figure 3 shows a visualisation of the results of classifying SNOMED augmented with the ontology from Figure 2. As can be seen, unifying generalisation concepts such as 84604002|sigmoid colectomy| have been identified, and thus provide a strong foundation for constructing queries that span the various procedures. Additionally, since SNOMED includes detailed anatomical concepts, queries can now be composed in terms of anatomical features even though they did not exist in the original terminology.

joins.

Procedure	Relation	SNOMED
32000-00	≡	315327002
32003-00	≡	315326006
32003-01	≡	235326000
32005-00	≡	43075005
32005-01	≡	174071004
32006-00	≡	82619000
32012-00	≡	26390003
32024-00	≡	400988008
32025-00	⊑	314592008
32026-00	⊑	314592008
32028-00	⊑	314592008
32030-00	≡	16564004
32039-00	≡	265414003
32051-00	⊑	174059005 ⊓ 70172002

Figure 2: Identified Relationships with SNOMED Concepts

COMPLEX QUERIES AND CONTEXT

So far we have only considered simple query scenarios where a single database column represents the concept we wish to query (e.g., `Surgery.procedure`) and there already exists a concept that characterises the bound of the query (e.g., 2396804).

Consider instead a table, as shown in Figure 4, that stores both scheduled and performed procedures while using another column to distinguish them, and which encodes laterality, if any, of the procedure in yet another column. Now imagine we wish to query for all patients who have had an amputation including the left hand.

Patient	Date	Status
...	...	...

	Procedure	Laterality
	...	...

Figure 4: Table storing records with contextual information split across columns

Patient	Date	...	Laterality	Code
...		...		...

Code	Equivalent SNOMED Expression
...	...

Figure 5: Augmented table for representing contextualised concepts

To support this kind of problem with reasonable generality and decent query speed, we need to generate a new column containing codes that are mapped to the set of compound concepts that correspond to the contextualised meaning of each database row. Hence, as shown in Figure 5, the table from Figure 4 would be extended with a `Code` foreign-key column, and an additional table containing the SNOMED expressions of the form[4]:

$$\exists \text{ associatedProcedure}.\langle \text{P} \rangle \quad \sqcap$$
$$\exists \text{ laterality}.\langle \text{L} \rangle \quad \sqcap$$
$$\exists \text{ procedureContext}.\langle \text{S} \rangle$$

which gives us another ontology extension that we can add to SNOMED.

Finally, in order to be able to pose a subsumption-based complex query involving composite concepts and have it evaluated at database join speeds, we can employ the same strategy: extend the ontology with a new fully-defined concept corresponding to our query expression, re-classify, and perform a join-based query using the new concept.

The need to construct compound expressions that explicitly represent the context associated with a record in a database occurs any time the data needs to be queried outside its original context. This may happen in as trivial a case as when one table in a database is joined with another, but the more general scenario occurs when integrating data from multiple data sources.

RESULTS

Classifying SNOMED

The practicality of creating local extensions of SNOMED is dependent on sufficient tool support and, as mentioned previously, a cornerstone of this is fast classification. Indeed we believe that near real-time feedback in an editing environment, be it an IDE for programming or a 3D architectural modelling tool, can have a transformational effect on the authoring and editing process.

To this end, we have implemented *snorocket*, using a slightly altered form of the algorithm in [5] written in Java. We use several optimised Map and Set data-structures tailored for ontologies with roughly the same number of concepts and roles as SNOMED. This implementation is able to classify

[4]Note that considerable experience with SNOMED and all its documentation may be required to construct suitable valid post-coordinated expressions like those above. Tool support for this is clearly an important issue and recent work in the IHTSDO Concept Model SIG on producing a Machine Readable Concept Model will be valuable for this.

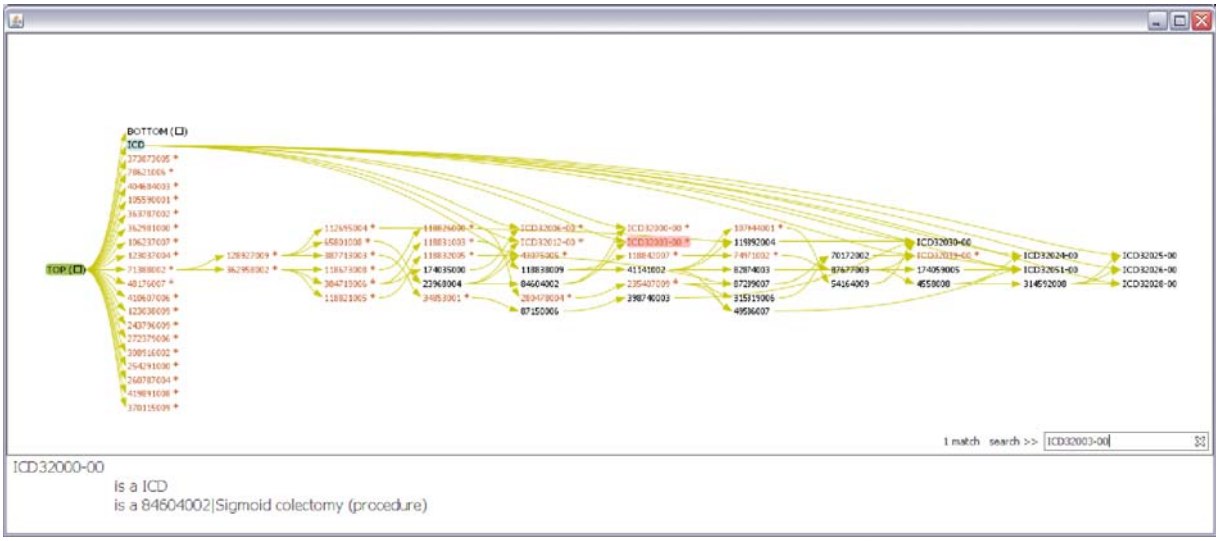

Figure 3: Visualisation of part of an extended SNOMED ontology

SNOMED in 54 seconds on a modern 2.4GHz Intel Core 2 Duo running Windows XP and Sun's Java 1.6.0_03.

For a fairer comparison with CEL, which only runs under Linux, we ran both snorocket and CEL on an older four-CPU Xeon 3.6GHz machine running RedHat Linux 2.6.9 and Sun's Java 1.6.0_04. The results, for several of the ontologies available from `http://lat.inf.tu-dresden.de/~meng/toyont.html`, are in Table 1.

Clearly, being able to classify SNOMED in close to a minute is a substantial improvement over roughly 23 minutes and brings us much closer to the near real-time feedback we are seeking.

Incremental Classification

In our mapping scenario we observe that SNOMED ($\mathcal{Q}$) is unchanging while the local extension ($\mathcal{P}$) is modified. If we can classify $\mathcal{Q}$ once and record the result $C(\mathcal{Q})$ then, due to the monotonicity of the description logic, the classification of $\mathcal{Q} \cup \mathcal{P}$, $C(\mathcal{Q} \cup \mathcal{P})$, is a superset of $C(\mathcal{Q})$. The goal is then to derive $C(\mathcal{Q} \cup \mathcal{P})$ given $C(\mathcal{Q})$ (and, of course, $\mathcal{Q}$ and $\mathcal{P}$) which should be much faster than deriving $C(\mathcal{Q} \cup \mathcal{P})$ from scratch.

Suntisrivaraporn [12] calls this Duo-Ontology Classification and presents a variation of the algorithm in [5] to do just this. We have independently derived our own variant of this algorithm along similar lines; the queue-processing core is essentially unchanged but the initialisation of the queues is different to account for the work that has already been done.

Currently this work is in a preliminary state and the correspondence with the variant described in [12] is unknown. However the performance of this incremental algorithm is very promising. With $\mathcal{P}$ consisting of the 14 new concepts as defined as in Figure 2, incremental classification takes around 0.9s using our un-optimised implementation.

DISCUSSION

Ideally, as a term set is developed, it would be explicitly constructed as an ontology and, to avoid re-invention and promote interoperability, could be developed as an extension of an existing standard ontology such as SNOMED. These extension ontologies could then be shared and evolved within their specialist community while still being useful and usable in more general communities. One such example is an ontology for skeletal dysplasias extracted from REAMS [13].

It is thus useful to be able to represent these ontologies in a standard format such as OWL so that they can be shared or manipulated using existing toolsets. Currently we use the OWL 1.1 proposal [14] rather than OWL 1.0 since it supports the expression of the role axioms (to describe role transitivity and right-identity). The particular subset we use is characterised by the description logic $\mathcal{EL}+^{\perp}$. OWL 1.1 is supported by, for example, the latest development-release of Protégé (4.0 alpha).

Unfortunately, OWL is not practical for representing large ontologies like SNOMED where an OWL

	SNOMED	FULL-GALEN	NOT-GALEN	NCI
CEL	1391.9	368.9	5.4	1.8
snorocket	72.8	15.1	0.4	0.4

Table 1: Comparison of classification time for snorocket and CEL running on the same hardware.

XML representation is approximately 240MB [15], about eight times the size of the equivalent KRSS representation. Moreover, due to the complexities inherent in parsing XML, it is much slower to load and parse than a simpler format such as KRSS.

One work-around for this, and something that would greatly benefit the e-health community, would be for the International Health Terminology Standards Development Organisation, the newly formed governing body of SNOMED, to formally publish URIs for the concepts in SNOMED. This would allow tool vendors to "bake in" SNOMED to their tools, while still allowing other OWL-based ontologies to reference SNOMED concepts in a consistent and interoperable manner in order to describe extensions to SNOMED.

CONCLUSION

Our preliminary work on producing local extensions to SNOMED for semantic data integration is promising as is the performance of our classifier. The current implementation is single-threaded and we anticipate a further speed increase from a multi-threaded implementation running on a multi-core CPU.

We are currently integrating snorocket with a 3rd-party SNOMED editing tool which requires specific support for SNOMED's use of role grouping and the ability to distinguish between stated and inferred relationships in the output of the classifier, although this adds little overhead to the classification time. In addition, we are prototyping mapping tools specifically targeting the task of constructing local extensions of SNOMED from existing data.

Finally, we are continuing work on our incremental form of the algorithm but have not yet tuned or verified the implementation. Preliminary results indicate that this approach should be very useable when integrated with our mapping tool.

Acknowledgements

The work described in this paper was carried out while on secondment to the CSIRO's E-Health Research Centre and the author would like to gratefully acknowledge the support of David Hansen and the other members of the Health Data Integration team.

Address for Correspondence

Michael J. Lawley, Faculty of Information Technology, University of Queensland, 126 Margaret Street Brisbane Qld 4000, Australia
m.lawley@qut.edu.au

References

[1] D. Hansen, C. Daly, K. Harrop, M. O'Dwyer, C. Pang, and J. Ryan-Brown. HDI: Research Software To Commercial Product. *ASWEC 2005 Industry Experience Papers*, 2005.

[2] I. Jakobsen, M.J. Lawley, A. Zankl, and D. Hansen. Ontologies for Skeletal Dysplasias. *MedInfo 2007 Workshop: MedSemWeb 2007*, 2007.

[3] SNOMED *Clinical Terms*. College of American Pathologists, 2006. `http://www.snomed.org`.

[4] F. Baader, C. Lutz, and B. Suntisrivaraporn. CEL—a polynomial-time reasoner for life science ontologies. In U. Furbach and N. Shankar, editors, *Proceedings of the 3rd International Joint Conference on Automated Reasoning (IJCAR'06)*, volume 4130 of *Lecture Notes in Artificial Intelligence*, pages 287–291. Springer-Verlag, 2006.

[5] F. Baader, C. Lutz, and B. Suntisrivaraporn. Efficient Reasoning in $\mathcal{EL}^+$. In *Proceedings of the 2006 International Workshop on Description Logics (DL2006)*, CEUR-WS, 2006. `http://lat.inf.tu-dresden.de/research/papers/2006/BaaLutSun-DL-06.pdf`.

[6] H. Wache, T. Vögele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. Hübner. Ontology-based integration of information-a survey of existing approaches. *IJCAI-01 Workshop: Ontologies and Information Sharing*, 2001:108–117, 2001.

[7] H. Stuckenschmidt. *Catalogue Integration: A Case Study in Ontology-based Semantic Translation*. Vrije Universiteit, Faculteit der Exacte Wetenschappen, Divisie Wiskunde & Informatica, 2000.

[8] Dieter Fensel, Ian Horrocks, Frank van Harmelen, Deborah L. McGuinness, and Peter F. Patel-Schneider. OIL: An Ontology Infrastructure for the Semantic Web. *IEEE Intelligent Systems*, 16(2), 2001.

[9] G. Wade and S.T. Rosenbloom. Experiences Mapping a Legacy Interface Terminology to SNOMED CT. In *Proceedings of the SMCS 2006 - Semantic Mining Conference on SNOMED CT*, 2006. `http://www.hiww.org/smcs2006/proceedings/9WadeSMCS2006final.pdf`.

[10] S. Ghilardi, C. Lutz, and F. Wolter. Did I Damage my Ontology? A Case for

Conservative Extensions in Description Logics. In Patrick Doherty, John Mylopoulos, and Christopher Welty, editors, *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR'06)*, pages 187–197. AAAI Press, 2006. http://lat.inf.tu-dresden.de/~clu/papers/archive/kr06a.pdf.

[11] *International Statistical Classification of Diseases and Related Health Problems, Tenth Revision, Australian Modification (ICD-10-AM)*. National Centre for Classification in Health, 5th edition, 2006. http://www3.fhs.usyd.edu.au/ncch/4.2.1.1.htm.

[12] Boontawee Suntisrivaraporn. Module extraction and incremental classification: A pragmatic approach for $\mathcal{EL}^+$ ontologies. In Sean Bechhofer, Manfred Hauswirth, Joerg Hoffmann, and Manolis Koubarakis, editors, *Proceedings of the 5th European Semantic Web Conference (ESWC'08)*, Lecture Notes in Computer Science. Springer-Verlag, 2008. To appear.

[13] C. Hall and J. Washbrook. Radiological Atlas of Malformation Syndromes and Skeletal Dysplasias (REAMS) [software]. Oxford University Press, CD-ROM, 1999.

[14] B. Motik, P.F. Patel-Schneider, and I. Horrocks. OWL 1.1 Web Ontology Language. World Wide Web Consortium, W3C Member Submission, 2006. http://www.w3.org/Submission/2006/SUBM-owl11-owl_specification-20061219/.

[15] K. Spackman. An Examination of OWL and the Requirements of a Large Health Care Terminology. In *Proceedings of the OWL: Experiences and Directions Third International Workshop (OWLED 2007), CEURWS*, June 2007. http://owled2007.iut-velizy.uvsq.fr/PapersPDF/submission_26.pdf.