A Description Logic Based Schema for the Classification of Medical Data

Ian Horrocks and **Alan Rector** and **Carole Goble**
Medical Informatics Group, Department of Computer Science, University of Manchester,
Oxford Road, Manchester, M13 9PL, UK
{horrocks—rector—carole}@cs.man.ac.uk

Abstract. The European Galen project aims to promote the sharing and re-use of medical data by providing a concept model which can be used by application designers as a flexible and extensible classification schema. A description logic style terminological knowledge representation system called Grail has been developed specifically for this task. Using a description logic based schema has a number of important benefits including coherence checking, schema enrichment and query optimisation.

In order to support a variety of design requirements Grail includes transitive closure of roles and general concept inclusions. Replacing the Grail classifier's existing structural subsumption algorithm with a sound, provably complete and decidable tableaux calculus based algorithm would have many attractions if the intractability problem could be mitigated by suitable optimisations. The optimisation of non-deterministic constraint expansion would be of particular importance as large numbers of these constraints can be introduced by general concept inclusions. Both intelligent back-tracking and the use of meta-knowledge to guide constraint expansion are being studied as possible methods of tackling this problem.

1 INTRODUCTION

A key aspect of linking medical databases is to harmonise their terminology which may run to 150,000 terms or more from any of more than two dozen major medical 'controlled vocabularies' of various types. The terminologies interact intimately with the database schemata, and many are specific to particular databanks. Most have been designed for statistical analysis or bibliographic retrieval, and additional ad hoc terms to cover the fine detail required for clinical care abound.

The very large size of static coding schemes makes them difficult to build and maintain. Schemes such as SNOMED which tackle this problem by allowing codes to be combined from a number of broad axes suffer from vague semantics, allowing single codes to have multiple interpretations and single concepts to have multiple codes. They are also often insufficiently constrained to prevent the potential generation of large numbers of nonsensical codes: T-67000+M-12000+E-4986+F-90000 is the SNOMED code for a fracture of the colon caused by donkey and emotional state [Nowlan,1993]. The limitations of static schemes has led to a proliferation of different systems biased towards different applications and areas of medical specialisation.

The European Galen project aims to promote the sharing and re-use of medical data by providing a concept model which can be used by application designers as a flexible and extensible classification schema [Rector *et al.*,1993]. By using a description logic to build a conceptual model it is hoped to avoid many of the pitfalls of existing static coding schemes as well as providing additional benefits to applications:

- more detailed descriptions with clear semantics can be constructed systematically to provide principled extensions to the terminology where required;
- the description logic classifier can be used to check the coherence of new descriptions and to enrich the schema by the discovery of implicit subsumption relationships;
- the description logic can be used as a powerful database query language supporting intensional as well as extensional queries [Bresciani,1995] and query optimisation [Beneventano *et al.*,1994];
- data can be shared between existing applications by using the concept model as an interlingua and providing mappings to a variety of coding schemes.

However, to achieve these aims requires coping with the basic structure of medical terminology which involves coordinating several taxonomies—a generic kind-of taxonomy, several different part-of hierarchies, and various causal relations. Anatomical and causal relations are fundamental to medical nomenclature. Any system which cannot cope with the basic fact that the "shaft of the femur" is a *part of* the "femur" but that a "fracture of the shaft of the femur" is a *kind of* a "fracture of femur" will not be satisfactory [Nowlan,1993].

Furthermore an interlingua must cope with bridging different levels of detail. For example, ulcers only occur on the lining of organs, so it is usually sufficient to record "ulcer of stomach". However, this expression needs to be classified correctly as the *same* concept as "ulcer of the lining of the stomach" and as an *ancestor* of "ulcer of the lining of the upper third of the stomach".

2 THE GRAIL DESCRIPTION LOGIC

Standard description logics[1] have been shown to cope poorly with the requirements of medical terminology [Doyle and Patil,1991] and so a new description logic, Grail [Goble *et al.*,1994], has been developed specifically for the task. Grail provides additional features:

[1] In particular the early description logic Nikl [Moser,1983], and by extension logics such as Back [Peltason,1991], Classic [Patel-Schneider,1991] and Loom [MacGregor,1991] which provide similarly restricted terminological reasoning.

- transitive closure of roles, supporting the coordination of multiple taxonomies based on relations other than subsumption, for example part-whole relations in the anatomical taxonomy;
- general concept inclusions, supporting the re-use of data across applications with varying requirements for descriptive detail;

In order to minimise tractability problems the design of GRAIL specifically excludes constructs which are deemed to be non-essential for the description of medical terminology [Nowlan,1993] and as a result the logic does not have negation, disjunction or general number restrictions[2]. However the addition of some or all of these constructs may be necessary if GRAIL is to be used in other problem domains. In particular the lack of number restrictions has proved to be a serious restriction when using GRAIL to describe and classify multimedia objects [Bechhofer and Goble,1996].

2.1 COORDINATING TAXONOMIES

GRAIL does not support specialised reasoning about part-whole relations and compositional inclusion [Padgham and Lambrix,1994] or about the interaction of different compositional relations [Sattler,1995]. What *is* provided is a general terminological mechanism called specialisation, which is similar to the assertional mechanism provided by the *transfers-thro* construct in CycL [Lenat and Guha,1989]. Specialisation allows the the user to specify that a characteristic is inherited across relations other than is-a: stating that role R is *specialisedBy* role S leads to the inference that for any objects x, y and z $xRy \wedge ySz \Rightarrow xRz$. This can be represented in the description logic using a combination of role composition and transitive reflexive closure by substituting the role $R \circ S^*$ for the role R. Note that the special case where a role is *specialisedBy* itself is equivalent to transitive closure.

Application of the specialisation mechanism is illustrated by the coordination of locative relations with the anatomical taxonomy. One requirement is that the *hasLocation* role transfers through the *isPartOf* role so that any x which *hasLocation* y which *isPartOf* z is classified as a kind of x which *hasLocation* z. Using $\mathit{hasLocation} \circ \mathit{isPartOf}^*$ instead of *hasLocation* gives the required subsumption inference:

$$\begin{gathered} \mathit{Fracture} \sqcap \exists(\mathit{hasLoc} \circ \mathit{isPartOf}^*).\mathit{Femur} \\ \text{subsumes} \\ \mathit{Fracture} \sqcap \exists(\mathit{hasLoc} \circ \mathit{isPartOf}^*). \\ (\mathit{Shaft} \sqcap \exists \mathit{isPartOf}.\mathit{Femur}) \end{gathered}$$

2.2 RE-USE OF DATA

If data is to be shared and re-used the schema must be adaptable to a wide range of application requirements. The use of a description logic based schema provides for principled extensibility while the use of concept inclusion axioms provides for varying levels of detail without compromising generality.

For example a concept inclusion axiom such as:

$$\begin{aligned} &\mathit{ulcer} \sqcap \exists \mathit{hasLoc}.\mathit{stomach} \\ &\sqsubseteq \quad \mathit{ulcer} \sqcap \exists \mathit{hasLoc}.(\mathit{lining} \sqcap \exists \mathit{isPartOf}.\mathit{stomach}) \end{aligned}$$

allows applications to use a less detailed description while still obtaining all the classification inferences associated with the full description. In addition the more detailed description can be recognised as defining the same set of objects as the minimal form.

3 THE GRAIL CLASSIFIER

As part of the GALEN project a GRAIL classifier has been implemented and a fragment of a medical terminology model (approximately 3,000 concepts) constructed. The classifier uses a structural algorithm for basic subsumption testing and an iterative recursive reclassification algorithm to deal with general concept inclusions. There are a number of problems with this architecture including incomplete reasoning[3], possible non-termination and limited extensibility. While some incompleteness might be acceptable, the behavior of the existing classifier is poorly understood and the missed inferences difficult to characterise. The relative difficulty of extending the set of concept forming operators has also proved to be serious impediment to the use of GRAIL in other problem domains.

Theoretical work has now shown that similar expressiveness could be provided by extending $\mathcal{ALC}$, either with the transitive closure of roles ($\mathcal{ALC}_{\text{TRANS}}$) [Baader,1990] or by allowing general concept inclusion axioms and terminological cycles [Buchheit *et al.*,1993]. Sound, complete and decidable subsumption testing is possible for these logics using tableaux calculus algorithms with enhanced control strategies. Tableaux algorithms have the additional attraction that it is fairly straightforward to add concept and role forming operators by extending the set of expansion rules, although some combinations of operators are known to lead to undecidability.

The major drawback to the use of sound and complete subsumption algorithms is the complexity which, for $\mathcal{ALC}$ extended with transitive closure and features has been shown to be exponential in time with respect to the size of the knowledge base [Schild,1991]. However this is a worst case result and reflects pathological constructs which occur rarely, if at all, in practice. It has also been shown that, with suitable optimisations, sound and complete algorithms can provide acceptable performance for basic $\mathcal{ALC}$ when used with realistic knowledge bases [Baader *et al.*,1992]. The approach which is proposed for GRAIL is to investigate optimisation techniques for more expressive description logics and, if these prove insufficient for acceptable performance, to retreat gracefully into incompleteness.

4 EXTENDED TABLEAUX SUBSUMPTION ALGORITHMS

In practice a serious cause of intractability is the use of general concept inclusions—when they are present in a knowledge base they must be considered in every subsumption test.

For example to decide if C subsumes D, tableaux calculus algorithms test the satisfiability of $D\sqcap\neg C$ by expanding a constraint system S, initialised to contain $\{x_1 : (D\sqcap\neg C)\}$, until it defines a model or reveals obvious contradictions which prove that there is no model. If the terminology contains concept inclusions of the form $A \sqsubseteq B$, they can be rewritten $B\sqcup\neg A \doteq \top$ and added to S as constraints of the form $\forall x.x : (B \sqcup \neg A)$ which means that the constraint is applied to every variable in S—variables represent objects in the domain all of which must, by definition, be in the extension of $\top$.

[2] Single-valued/functional roles are provided.

[3] In common with most other structural subsumption algorithms.

The disjunctive form of concept inclusion constraints means that the expansion of S will be non-deterministic and could lead to the exploration of multiple constraint systems. Even if the expansion of S does not produce any exists-in constraints, so no new variables are created, a set of n concept inclusions could lead to 2^n different constraint systems being explored in the worst case; if the expansion leads to the creation of $(m-1)$ new variables this rises to 2^{nm} in the worst case.

Moreover the worst case, or at least a seriously bad case, is likely to arise when $D \sqcap \neg C$ is not satisfiable with respect to one or more of the concept inclusions. For example a concept term $D \sqcap \neg C$ is obviously not satisfiable with respect to a terminology which contains the concept inclusion $D \sqsubseteq C$ as this would lead to a constraint system $S = \{x_1 : (D \sqcap \neg C), \forall x.x : (C \sqcup \neg D), \ldots\}$. If there are a large number of other concept inclusion constraints, none of which cause a clash with $x_1 : (D \sqcap \neg C)$, the order in which the constraints are expanded will dictate the number of constraint systems which are explored before the non-satisfiability is discovered.

This problem is exacerbated by the control strategy used to ensure termination of the algorithm when concept inclusions are supported [Buchheit *et al.*,1993]. The strategy dictates that constraints on a variable x_i must be fully expanded before constraints on successor variables are expanded and that exists-in constraints on x_i are the last to be expanded. This means that if the non-satisfiability of a constraint system is only demonstrated by a clash on variable x_{i+1}, a naive application of the control strategy would ensure worst case behavior with respect to x_i and all its predecessors. Consider for example a terminology $\mathcal{T}$ consisting entirely of concept inclusions:

$$\begin{aligned} \mathcal{T} \quad = \quad & \{\forall R.\neg D \sqsubseteq \exists R.C, \\ & A_1 \sqsubseteq B_1, \\ & \vdots \\ & A_n \sqsubseteq B_n\} \end{aligned}$$

Testing if $\forall R.\neg D$ is subsumed by $\exists R.C$ with respect to $\mathcal{T}$ leads to the expansion of the constraint system:

$$\begin{aligned} S \quad = \quad & \{x_1 : (\forall R.\neg D \sqcap \forall R.\neg C), \\ & \forall x.x : (\exists R.C \sqcup \exists R.D), \\ & \forall x.x : (B_1 \sqcup \neg A_1), \\ & \vdots \\ & \forall x.x : (B_n \sqcup \neg A_n)\} \end{aligned}$$

Although it is clear that $x_1 : (\forall R.\neg D \sqcap \forall R.\neg C)$ and $x_1 : (\exists R.C \sqcup \exists R.D)$ will always cause a clash when $\{x_1 R x_2\}$ and either $\{x_2 : C, x_2 : \neg C\}$ or $\{x_2 : D, x_2 : \neg D\}$ are added to S the control strategy will ensure that all 2^{n+1} possible expansions of the $\forall x.x$ constraints on variable x_1 are explored before this is discovered.

5 OPTIMISATION TECHNIQUES

Tractable classification for realistic terminologies containing significant numbers of concept inclusions will require optimisation of the tableaux expansion algorithm to deal more effectively with the disjunctive constraints they introduce. Techniques being investigated include more intelligent exploration of alternative constraint systems and minimisation of the number of disjunctive constraints which must be expanded.

5.1 INTELLIGENT BACKTRACKING

The performance of the tableaux expansion algorithm could be greatly improved by more intelligent backtracking when a clash is discovered after the expansion of one or more disjunction constraints. By default the algorithm will explore the alternatives offered by the most recently expanded disjunction and, if none of them leads to a clash free constraint system, backtrack one disjunction constraint at a time until a clash free system is discovered or all possible alternatives have been explored. As described in section 4, this can lead to wasted exploration when existing constraints lead deterministically to a clash.

To deal with this problem constraints can be marked to indicate when they stem from a non-deterministic expansion and, if so, which one. When a clash is detected it will then be possible to backtrack directly to the most recently expanded disjunction which offers the possibility of eliminating one or both of the clashing constraints.

This technique should also result in the rapid detection of a clash resulting from the deterministic expansion of an initial constraint system and thus deal efficiently with the case where $D \sqcap \neg C$ is inherently unsatisfiable.

5.2 USING META-KNOWLEDGE

An example of this technique is an optimisation method which uses knowledge of the structure and function of general concept inclusions in the GALEN terminology:

- In inclusion axioms $A \sqsubseteq B$, A is always a conjunctive concept which can be expanded so that one of the conjuncts is a primitive concept;
- Concept inclusions usually represent additional intensional knowledge about some specific concept and will affect only a small proportion of subsumption tests;
- The 'top' of the model consists of an extensive primitive hierarchy which is largely disjoint—few of the primitive concepts have multiple parents.

Equation (1) is a typical general concept inclusion statement from the GALEN model; it represents the knowledge that a high lymphocyte count is a pathological condition. The interaction of this concept inclusion and the transfers-through mechanism with respect to causation and pathology results in conditions which *cause* high lymphocyte counts being classified as pathological conditions. As a result of this concept inclusion the constraint (2) would appear in all constraint systems used for subsumption testing.

$$\begin{aligned} & \mathit{LymphocyteCount} \sqcap \exists \mathit{level.high} \\ & \qquad \sqsubseteq \exists \mathit{status.pathological} \end{aligned} \qquad (1)$$

$$\begin{aligned} & \forall x.x : (\exists \mathit{status.pathological} \sqcup \neg \mathit{LymphocyteCount} \\ & \qquad \sqcup \forall \mathit{level}.\neg \mathit{high}) \end{aligned} \qquad (2)$$

Given that the knowledge represented in this concept inclusion is only relevant to a small proportion of subsumption tests—those which refer to lymphocyte counts—it would make sense to preferentially select the $\neg$*LymphocyteCount* term when expanding the disjunctive constraint. Furthermore the concept *LymphocyteCount* is defined by a conjunction which contains the primitive concept *CountConcentration* so the constraint could be expanded to give:

$$\begin{aligned} & \forall x.x : (\exists \mathit{status.pathological} \sqcup \neg \mathit{CountConcentration} \sqcup \\ & \qquad \ldots \sqcup \forall \mathit{hasLevel}.\neg \mathit{high}) \end{aligned}$$

When expanding this constraint on a variable x_i in a constraint system S, preferentially selecting the expansion:

$$S \cup \{x_i : \neg \mathit{CountConcentration}\}$$

is likely to be a good choice as the new constraint requires no further expansion[4] and will only cause a clash when x_i : *CountConcentration* is in S.

A set of n concept inclusions $A_1 \sqsubseteq B_1, \ldots, A_n \sqsubseteq B_n$, where all A_j are of the form $P_j \sqcap a_{j1} \sqcap \ldots \sqcap a_{jk}$ and P_j is a primitive concept, can be expanded to give n constraints of the form $\forall x.x : \neg P_j \sqcup \neg a_{j1} \sqcup \ldots \sqcup \neg a_{jk} \sqcup B_j$. The tableaux expansion algorithm can then make the default assumption that all variables x_i in S are subject to the constraints $x_i : \neg P_1, \ldots, x_i : \neg P_n$. Only if $x_i : P_j$ is added to S will it be necessary to fully expand the $\forall x.x : \neg P_j \ldots$ constraint on x_i.

This is achieved by sorting the $\forall x.x : \neg P_j \ldots$ constraints into a table indexed by the primitives $P_1 \ldots P_n$[5] and checking the table whenever a constraint $x_i : P$ is added to S. If P is found in the table the corresponding $\forall x.x : \neg P_j \ldots$ constraints must then be applied to x_i[6].

As there is an extensive and largely disjoint primitive hierarchy at the top of the GALEN terminology, restricting expansion of concept inclusion constraints in this way allows the majority of 'irrelevant' inclusion constraints to be dealt with efficiently without compromising completeness. Preliminary experiments indicate that, on average, less than 10% of concept inclusion constraints in the GALEN model will need to be fully expanded in a typical subsumption test.

6 CONCLUSION

A flexible and extensible classification schema for use by application designers would facilitate the inter-operability of medical databases and promote data sharing and re-use. Using a description logic for schema design and maintenance would provide automatic coherence checking and the enrichment of the schema through the discovery of implicit subsumption relations. The description logic could also act as a powerful query language offering both query optimisation and the ability to answer intensional as well as extensional queries. Finally, by acting as an interlingua, a description logic based schema could provide a mechanism for the exchange of data between legacy systems which use a variety of existing coding schemes.

Coping with the complexities of medical terminology across a wide range of applications will however require a highly expressive description logic. Using such a description logic with any but the smallest knowledge base will require effective optimisation and, in particular, optimisation of the expansion of constraints introduced by general concept inclusion axioms. Using meta-knowledge is one possible approach and, although work is still at a very preliminary stage, it appears to show some promise. These kinds of optimisation are particularly effective in the GALEN knowledge base due to the predominance of conjunction concepts and the high level of disjointness among primitive concepts. However we believe that these characteristics are common to many knowledge bases in real problem domains.

Even if the most highly optimised sound and complete procedure proves to be too inefficient to be used in applications, the existence of such a procedure has a number of benefits [Buchheit *et al.*,1993]: it provides a benchmark for judging incomplete procedures; it could be used for 'background' processing after a quick answer has been provided by an incomplete procedure; and it is a sensible starting point from which to retreat gracefully into limited and clearly characterised incompleteness.

References

[Baader *et al.*, 1992] F. Baader, B. Hollunder, B. Nebel, and H.-J. Profitlich. An empirical analysis of optimization techniques for terminological representation systems. In B. Nebel, C. Rich, and W. Swartout, editors, *Principals of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR '92), Cambridge, MA*, pages 270–281, Morgan-Kaufmann, San Mateo, CA, October 1992. Also available as DFKI RR-93-03.

[Baader, 1990] F. Baader. Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles. Research Report RR-90-13, Deutsches Forschungszenrum für Künstliche Intelligenz GmbH (DFKI), 1990.

[Bechhofer and Goble, 1996] S. Bechhofer and C. Goble. Description logics and multimedia—applying lessons learnt from the GALEN project. In *Proceedings of the workshop on Knowledge Representation for Interactive Multimedia Systems (KRIMS'96), at ECAI'96, Budapest, Hungary*, August 1996. To appear.

[Beneventano *et al.*, 1994] D. Beneventano, S. Bergamaschi, S. Lodi, and C. Sartori. Terminological logics for schema design and query processing in OODBs. In F. Baader, M. Buchheit, M.A. Jeusfeld, and W. Nutt, editors, *Reasoning about structured objects—knowledge representation meets databases. Proceedings of the KI'94 Workshop KRDB'94, Saarbrücken, Germany*, September 1994.

[Bresciani, 1995] P. Bresciani. Querying databases from description logics. In F. Baader, M. Buchheit, M.A. Jeusfeld, and W. Nutt, editors, *Reasoning about structured objects—knowledge representation meets databases. Proceedings of the 2nd Workshop KRDB'95, Bielefeld, Germany*, September 1995.

[Buchheit *et al.*, 1993] M. Buchheit, F. M. Donini, and A. Schaerf. Decidable reasoning in terminological knowledge representation systems. *Journal of Artificial Intelligence Research*, 1:109–138, 1993.

[Doyle and Patil, 1991] J. Doyle and R. Patil. Two theses of knowledge representation: Language restrictions, taxonomic classification, and the utility of representation services. *Artificial Intelligence*, 48:261–297, 1991.

[Goble *et al.*, 1994] C. A. Goble, S. K. Bechhofer, W. D. Solomon, A. L. Rector, W. A. Nowlan, and A. J. Glowinski. Conceptual, semantic and information models for medicine. In *Proceedings of the 4th European-Japanese Seminar on*

[4] A constraint $x_i : \neg P$, where P is a primitive concept, requires no further expansion even if subsumption relations $P \sqsubseteq C_1 \ldots PsqsubseteqC_n$ have been asserted. In such a case $P \doteq P' \sqcap C_1 \sqcap \ldots \sqcap C_n$, where P' is a unique atomic primitive, and $\neg P \doteq \neg P' \sqcup \neg C_1 \ldots \sqcap C_n$. A clash can only be caused if $x_i : P' \in S$ which is only possible if $x_i : P \in S$.

[5] If concept names are mapped to numbers this is a simple array lookup; otherwise a hash table can be used.

[6] In fact it is only necessary to apply the constraint $\forall x.x : \neg a_{j1} \sqcup \ldots \sqcup \neg a_{jk} \sqcup B_j$ as it has already been discovered that $x_i : \neg P$ would cause a clash

Information Modelling and Knowledge Bases, Stockholm, 31st May – 3rd June 1994.

[Lenat and Guha, 1989] D. B. Lenat and R. V. Guha. *Building Large Knowledge-Based Systems*. Addison-Wesley, 1989.

[MacGregor, 1991] R. M. MacGregor. Inside the LOOM description classifier. *SIGART Bulletin*, 2(3):88–92, 1991.

[Moser, 1983] M. G. Moser. An overview of NIKL: the new implementation of KL-ONE. Technical report 5421, Bolt, Beranek and Newman, Cambridge, MA, 1983.

[Nowlan, 1993] W. A. Nowlan. *Structured methods of information management for medical records*. PhD thesis, University of Manchester, 1993.

[Padgham and Lambrix, 1994] L. Padgham and P. Lambrix. A framework for part–of hierarchies in terminological logics. In J. Doyle, E. Sandewall, and P. Torasso, editors, *Principals of Knowledge Representation and Reasoning: Proceedings of the Fourth International Conference (KR '94), Bonn, Germany*, pages 485–496, 1994.

[Patel-Schneider, 1991] P. F. Patel-Schneider. The CLASSIC knowledge representation system: Guiding principals and implementation rationale. *SIGART Bulletin*, 2(3):108–113, 1991.

[Peltason, 1991] C. Peltason. The BACK system—an overview. *SIGART Bulletin*, 2(3):114–119, 1991.

[Rector *et al.*, 1993] A. L. Rector, W A Nowlan, and A Glowinski. Goals for concept representation in the GALEN project. In *Proceedings of the 17th Annual Symposium on Computer Applications in Medical Care (SCAMC'93), Washington DC, USA*, pages 414–418, 1993.

[Sattler, 1995] U. Sattler. A concept language for engineering applications with part–whole relations. In *Proceedings of the International Conference on Description Logics—DL'95, Roma, Italy*, pages 119–123, 1995.

[Schild, 1991] K. Schild. A correspondence theory for terminological logics: Preliminary report. In *Proceedings of IJCAI'91*, pages 466–471, 1991.