

A Terminological Logic with Defaults: A Definition and an Application]*

Lin Padgham and Tingting Zhang
Department of Computer and Information Science
Linköping University, S-581 83 Linköping, Sweden.
lin@ida.liu.se, tzh@abacus.hgs.se

Abstract

In this paper we present a terminological language which includes defaults, and a definition of default subsumption based on the notion of skeptical inheritance in default reasoning. Except for the inclusion of defaults the language is limited when compared to most terminological logics. However defaults are a necessary construct in many applications and we suggest that the language presented here is a useful tradeoff between different types of expressivity. We present an algorithm for classifying new concepts into the default hierarchy representing the taxonomy, and in addition an algorithm for what we call default classification, suitable for interactive reasoning about individuals. We describe a diagnosis application which has been implemented using this language and reasoning mechanisms. We present an evaluation of the diagnosis application on the basis of comparison with 63 patient protocols. We conclude that the language presented is in fact adequate for the application presented here and hypothesize that it is interesting for a significant group of applications.

1 Introduction

Taxonomical representations and reasoning have been popular in A.I. for a long time, largely because they provide an efficient way of representing knowledge and organising it for reasoning purposes. The two main kinds of taxonomical reasoning to be found in the literature are classification (e.g. [Brachinan *et al.*, 1991]) and default inheritance reasoning (e.g. [Horty *et al.*, 1987; Fahnan, 1979]). Classification in systems with strict inheritance hierarchies has been used in a number of applications. However the kind of classification systems used here have also been criticised for not having the necessary expressivity of defaults [Doyle and Patil, 1991]. Default reasoning on the other hand has been used very little in real applications since the early work with NETL [Fahlman, 1979].

Recently there has been increased interest in incorporating defaults and principled default reasoning reason-

*This work is supported by funds from the Swedish Institute for Technological Development,

ing into classification systems. Some problems have been identified [Padgham and Nebel, 1993] and some theories for integrating default reasoning in terminological logics¹ have been developed [Padgham and Nebel, 1993; Quantz and Royer, 1992; Baader and Hollunder, 1992]. In this work we define a language which is relatively simple compared to most terminological logics (e.g. [Patel-Schneider *et al.*, 1990]), but which also contains important constructs not (usually) found in other languages of this type - namely defaults and features with predicate values. (The latter is however not the concern of this paper). In the search for computationally manageable but sufficiently expressive languages to attack real world problems [Donini *et al.*, 1991] we hypothesize that this sort of language meets the needs of many applications - for instance some applications in natural language and certain kinds of medical applications.

We have used the language and associated reasoning mechanisms to build an interactive medical diagnosis system in the sub-domain of diseases of the genito-urinary system. Although this is a limited domain it is one in which diagnosis is a genuine problem [Timpka, 1989]. The knowledge base is built by taking information directly from medical texts [Wor, 1977; Findel *et al.*, 1981; Braunwald *et al.*, 1981] and using our defined language, representing this information as a default inheritance taxonomy.

We have evaluated this system by using material from a medical study on diagnosis of patients with genito-urinary diseases.² The evaluation shows our system as behaving comparably to doctors on the available patient sample.

The following sections describe the language and the associated reasoning mechanisms which have been developed in the context of the diagnosis application. The application is presented briefly and the results of an empirical evaluation are summarised.

2 The DTL Language

In designing the language to be used we begin with the default theory of [Padgham, 1989]. This theory relies on

Alternative names are description logics or KL-ONE like.
²The protocols were collected by Dr Toomas Timpka, and are discussed in [Timpka, 1989].

```

< DTL > ::= {< terminology > | < default-reachable >}*
< terminology > ::= {< concept-introduction > \ < restriction >}*
< concept-introduction > ::= {< atomic-concept > = < concept > \ < atomic-concept > < < concept > >}
< concept > ::= Anything | < atomic-concept > | (And < concept >+) |
              (ExistAll < feature > < predicate >) | (All < feature > < predicate >)
< feature > ::= < identifier >
< predicate > ::= < relop > < real > | in < set-domain >
< relop > ::= < | > | < = | > = | =
< set-domain > ::= < string >*
< atomic-concept > ::= < probable > (< identifier >)
< probable > ::= core | default
< restriction > ::= (disjoint < atomic-concept > < atomic-concept >)
< default-reachable > ::= (default-reachable < atomic-concept > < concept >)

```

Figure 1: BNF definition of the DTL language

the notion that each type is represented by two nodes in an inheritance graph - a default node representing the typical properties of the type, and a core node representing the necessary properties of the type. Consequently we introduce the notion of an atomic type being either a core or a default. An important semantic constraint is that the core of a concept is always subsumed by the default of that concept.

The syntax of DTL follows the style used in [Nebel, 1990] and is given in figure 1. We note that there are no roles (relational constructs), but that we instead allow features with numerical ranges or sets of strings as values. This is extremely important in the medical domain (e.g. RBC-count \geq 50). Features with numeric values add significant expressivity and are not problematic for the subsumption algorithms.

Due to space restrictions we do not here define the formal semantics of the language. However, with the exception of the default reachability construct, the language can be defined in the usual set-theoretic way [Nebel, 1990] with the addition of an axiom which states that the extension of the default is always a superset of the extension of the core. The default reachability construct is an outcome of a default reasoning process applied to the terminology. This process is intensionally, rather than extensionally defined. The default-reachability process which we employ is that defined in [Padgham, 1989] as skeptical default inheritance. This process gives for any node (or ANDed combination of nodes) in the terminology, a set of nodes which are considered to be default-reachable. This set is essentially those conclusions sanctioned by the skeptical extrusion defined by e.g. [Horty et al., 1987].³

Although the DTL language is richer than typical inheritance network formalisms, it is nevertheless possible to capture it using only a slightly extended network language. We base our network language on that used in [Padgham, 1989] which uses core and default nodes (represented by a letter with subscripts "d" and "c" respectively), plus strict positive links " \Rightarrow " and strict negative links " \nRightarrow ". To this we add the notion of defined nodes (core or default) and property nodes. Defined nodes are

³Subtle differences are explained and motivated in [Padgham, 1989].

represented by an encircled letter plus subscript and are assumed to be defined by the conjunction of all nodes reachable by a single strict positive link. Property nodes are represented by a letter enclosed by a square, and represent an (All...) or an (ExistAll...) clause. This network language can also (usually)⁴ be mapped into a form using both strict links and default links (written " \Rightarrow " and " \rightarrow ") from a combined core/default node. This alternative form will be used in some examples as it requires fewer nodes and is therefore sometimes simpler to understand.

3 Reasoning Mechanisms

In order to have a terminological logic system, as opposed to simply a language we must also define the reasoning mechanisms which can operate on this language. Classification is the primary reasoning facility offered by T.L. systems, with subsumption being the most significant aspect of classification. We define a default subsumption relation, which together with strict subsumption can be used for classifying new concepts into the hierarchy. We also describe a method suitable for doing some limited reasoning about individuals with respect to this taxonomy.

3.1 Subsumption

At the centre of classification is the notion of subsumption. By testing whether the description to be classified subsumes or is subsumed by certain concepts in the network we establish the set of most specific subsumers and most general subsumees. Strict subsumption in our system is defined in an identical way to subsumption in standard terminological languages (e.g. [Nebel, 1990]). Subsumption between features is defined in the obvious way (e.g. "blood-count > 50" subsumes "blood-count > 60").

In addition to strict subsumption which is quite straightforward we also want to define a default subsumption relationship which takes into account the special semantic and structural relationship between the default and the core of a concept. In order to do this we

⁴The only case where the mapping does not work is when we have a relationship originating in a default node. This extra expressivity is not needed in the examples in this paper.

rely on the default reachability relationship determined by the default inheritance reasoner.

The definition of default subsumption between atomic concepts is as follows:

Definition: For two atomic concepts X, Y , X *default-subsumes* Y iff X is default reachable from Y .

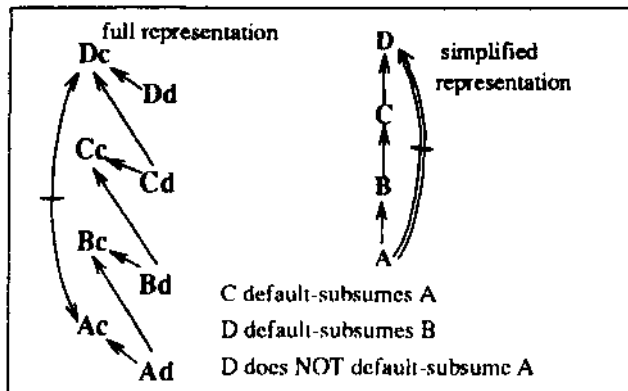


Figure 2: Illustration of default subsumption.

The full process of determining default-subsumption between arbitrary concept definitions is then analogous to that used in standard subsumption algorithms (e.g. [Nebel, 1990]), where the concept definition is unfolded and normalised and each part of the concept definition (in our case ANDed concepts. All clauses, ExistAll clauses and feature definitions) is tested separately. The only difference is that, for comparison of atomic concepts default-subsumption rather than strict subsumption is used.

The intuition behind default-subsumption is that if the object being classified is fully typical, then all default-subsumption relationships will also be subsumption relationships. If a strict inheritance hierarchy is coerced so that all the inheritance relations became defaults, default-subsumption on this network should give the same relationships as subsumption did on the original strict network.

3.2 Classification

The role of classification in terminological logic systems is to correctly place new concepts into the taxonomy so that they can be used for reasoning. We describe briefly the classification process used for adding new core and default concepts into our system.

The basic steps in a standard classification algorithm can be described as follows: X is the concept to be classified.

1. Traverse hierarchy collecting most specific subsumers of X .
2. Traverse hierarchy starting at children of one of the most specific subsumers, collecting most general subsumees of X .
3. Link X as child of most specific subsumers.
4. Link X as parent of most general subsumees.
5. Modify parent and child links in most specific subsumers and most general subsumees, if necessary.

Our algorithm is identical to this at a high level. However the definition of which are most specific subsumers (and most general subsumees) differs in our system. Just as in standard systems subsumers of X are those concepts which (strictly) subsume X . In a standard system the *most specific* subsumers of X , are those subsumers which do not subsume some other subsumer of X . In our system the most specific subsumers of a *default concept* are those which do not *default-subsume* some other subsumer. For a core concept the most specific subsumers are defined in the standard way. This allows default definitions to rely on default inheritance, while core definitions require guaranteed inheritance of properties.

Similarly the most general subsumees of a concept are defined to be those which are not default-subsumed by any other subsumee.

The rationale for using strict rather than default subsumption to determine which are the most specific subsumers for a core concept is that core concepts contain (by definition) strictly necessary properties. Thus the representation should ensure that these properties (or inheritance relationships) cannot be overridden.

Figure 3 shows an example of classification of a core and a default concept. 3a shows the links which would be inserted if the concept description was introduced as core(E), while 3b shows the situation that would arise with the introduction of the same concept description as default(F). In both cases the new concept description is (strictly) subsumed by default(A) and core(B). In the case of introducing a new core concept these links are simply added as there is no strict subsumption relationship between default(A) and core(B). However when the concept being introduced is a default only the link to default(A) is added as core(B) default-subsumes default(A), and is thus not regarded as one of the most specific subsumers. In the resulting taxonomy E objects will *always* inherit x from core(B) and z from core(A), via default(A) and core(E), and y from default(B) via core(E). F objects will by default inherit these same properties, via default(F).

The modification of parent and child links in most specific subsumers and most general subsumees follows the same principle as described above.

We note that if one adds information which in some way changes already defined concepts, then both strictly-subsumed and default-subsumed children can be affected. This is a common problem in terminological systems but is beyond the scope of the current paper.

3.3 Reasoning about Individuals

Classification is often used not only to classify new concepts but as an important component in the reasoning about individuals. The task here is to establish which classes the individual belongs to. In accordance with the use of defaults we want to establish which classes the individual *probably* belongs to. We call this *default classification*. This process actually requires interaction between default reasoning and classification which is non-trivial (see [Padgham and Nebel, 1993]). However a simpler approach can be suitable for either interactive applications where wrong assumption of defaults can be

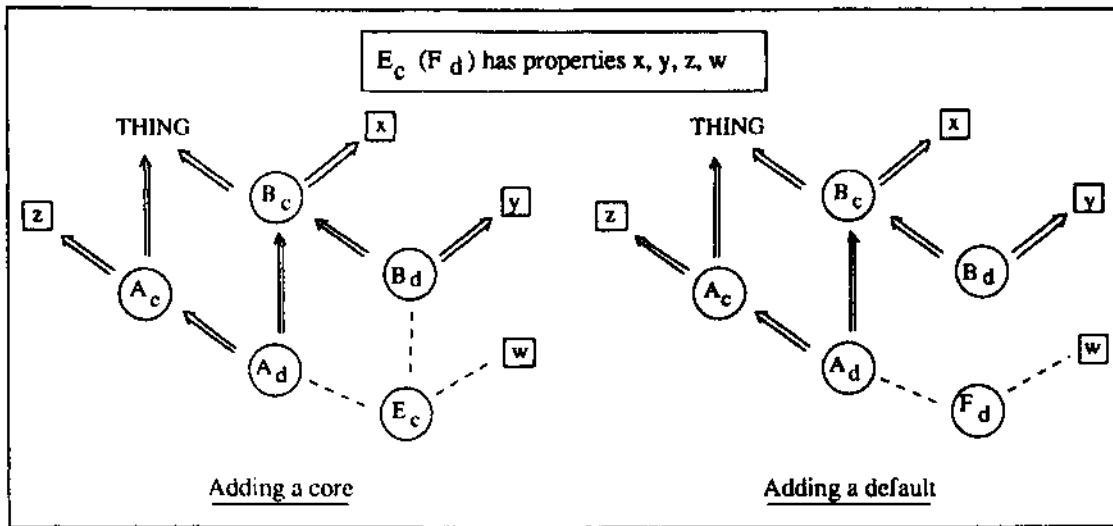


Figure 3: Classification of a new concept - core and default

discovered interactively (as in our diagnosis application) or for applications where the structural nature of conflicts is simple.⁵

In default classification of individuals we rely more heavily on default subsumption than we do in concept classification. As the hierarchy is traversed we collect all default-subsumers of the individual. (In concept classification we collected only strict subsumers). This ensures that, the reasoning process is pushed further down in the hierarchy, making use of default knowledge. However it incorporates only one level of default assumptions. In order to have full default, classification we would need to continue further from the result of this step, making further default assumptions on the basis of our default classification, followed by further approximate default classification and default reasoning until we reached a (possibly inconsistent) fix-point. Conflicts would then need to be resolved. However for the purposes of this paper we limit ourselves to the initial approximation which is computationally and conceptually simpler and is sufficient for the application discussed

4 The Diagnosis Application

We now describe our diagnosis application and the way in which it used the language and reasoning mechanisms described. Many domains represent their knowledge in a hierarchical or semi-hierarchical form e.g. medical textbooks are often organised according to disease hierarchies. If we assume that we have hierarchically organised domain knowledge and that a problem can be described in the language of the knowledge base, then diagnosis can be seen as conceptually quite similar to classification. If the problem description is complete we can find the set of possible diagnoses which are subsumed by the problem, and then take the minimum cover of

this set. For example in figure 4 the diagnosis would be "chronic pyelonephritis and acute cystitis".⁶ The assumption in taking the minimum cover is here that a diagnosis must explain (or cover) all aspects of the disease description. However it should also be a minimal explanation so we do not simply take the entire set of subsumees as a diagnosis.

The notion of diagnosis as essentially a classificatory process is intuitively appealing. It appears logical and it uses a taxonomical representation of the domain which we know is a common way for humans to organize knowledge. It also explains one general purpose reasoning strategy, in terms of another simpler such strategy — diagnosis is classification plus minimum cover.

However if we now move from the ideal case to a more real-world situation, there are a number of problems with this approach, although the basic model is still promising. The two major difficulties are the following:

- Problem descriptions in the real world are seldom complete.
- Taxonomies in the real world are seldom strict hierarchies. They are thus not a suitable representation for traditional classification approaches.

The latter of these two problems we address by using the language and reasoning mechanisms described above. The former problem we address by the notion of an interactive diagnosis system where the system seeks further information in a principled and directed way. We make an approximate and tentative default classification of the problem description and then ask questions to determine the value of features contained in subsumee nodes but, unknown in the problem description. We use techniques from information theory in order to ask first those questions whose answers will best discriminate between possibilities. This latter technique we hypothesize

⁵Due to space limitations we cannot here go into details about the nature of the structural constraints required.

⁶The disease descriptions used here are incomplete - both to save space and to simplify.

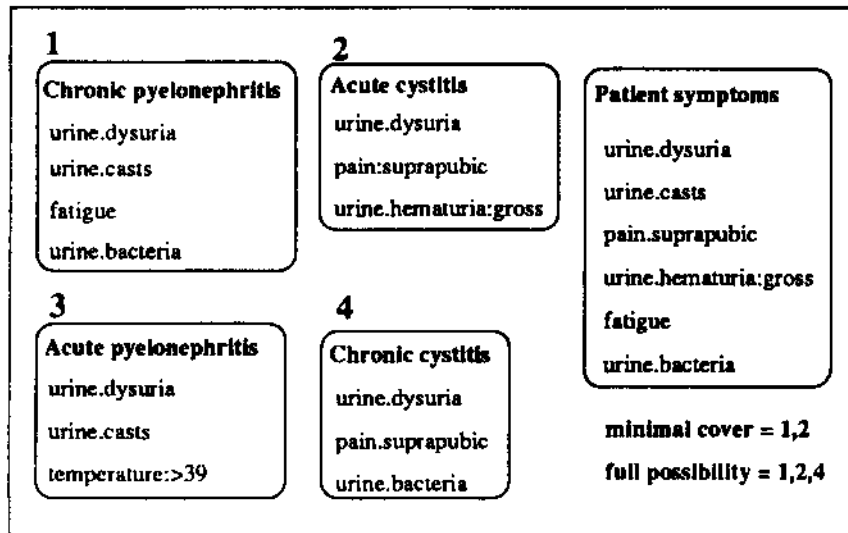


Figure 4: The diagnosis should be a minimal cover of the problem

may well simulate the "compiled knowledge" of experts who "know" which are the most telling questions to ask or lab tests to request. A detailed discussion of the mutual information analysis is however beyond the scope of this paper. By rerunning our approximate default classification algorithm following the collection of new information we ensure that initial (possibly) faulty assumptions are modified as the process progresses.

4.1 The Knowledge Base

The knowledge base was built using the *DTL* language. The knowledge base is thus a default inheritance hierarchy over disease classes of the form used in [Padgham, 1989; Padgham, 1992]. This information was taken directly from the standard medical text [Wor, 1977].

Each disease is characterised by a set of properties describing such things as symptoms and results on lab tests. Each property was determined to be *typical, necessary or possible.* The properties with which diseases were annotated were taken from [Braunwald *et al.*, 1981; Findel *et al.*, 1981]. A medical doctor checked the representation and in particular indicated whether properties were typical, necessary or possible.

All diseases were entered as primitive concepts (i.e. using the < form for concept introduction). The rationale for this was that the represented information about a disease cannot be expected to fully define that disease. As a result of this the classifier never added new subsumption links — however it did check for consistency and maximize inheritance.

Possible properties are not part of the original theory of [Padgham, 1989], or of the language defined in figure 1. However they were found to be necessary for this application and were added to the language simply as an alternative form for the < *probable* > construct. A discussion of why these were necessary and how they can be integrated with the original theory can be found in [Zhang, 1993].

4.2 The Diagnostic Engine

The diagnosis in our system is a non-monotonic interactive process consisting of two main parts. There is first the approximate default classification of the presenting problem and then an analysis regarding which information it is most relevant to request.⁸ The expectation is that addition of the new information will enable specialisation of the initial classification on the next time through the loop. However it is also possible that the new information will lead to different default subsumption relationships and an orthogonal classification, thus making the process non-monotonic in nature. Eventually the classification reaches a single terminal node in the disease hierarchy (or in the case of multiple diseases, a set of terminal nodes, none of which is sufficient in itself) and the diagnosis is complete.

In the application one significant change was made to the approximate default classification algorithm as described in section 3.3. We note that as all of the concepts are declared as primitive, none of them can actually subsume the problem description unless they are explicitly or implicitly declared to do so. However this assumption appears to be inappropriate for the use we are making of classification in this application. We hardly want to require that we can only make diagnoses which we have explicitly stated in the problem description! Consequently in our use of approximate default classification we disregard the fact that concepts are declared as primitive, and treat them as defined.

Once we have established which classes the problem description probably belongs to, We use this information to determine which concepts to consider in seeking further information about the problem. We build what we call a *pursuit set* of nodes which could reasonably be

⁸This analysis also relies on a knowledge base representation which uses defaults, but the details of these steps are beyond the scope of this paper. For full details see [Zhang, 1993]

pursued. We then analyse these taking into account certain heuristics (e.g. that a problem is more likely to be a typical disease manifestation than an atypical one) and develop an ordered list of information to be sought.

5 Empirical Evaluation of System

One way to evaluate our system is to ascertain how often it obtains the correct diagnosis. However it is not always easy to say what is a correct diagnosis. We therefore compared our system to medical doctors' diagnosis using patient protocols from 63 patients. The patients were seen by a general practitioner at a community health centre and forms were filled in regarding symptoms, signs, lab tests, etc. The GP's diagnosis was also recorded. The 63 patient description protocols (without diagnosis) were then given to each of 9 different specialists who were asked to make a diagnosis on the basis of the recorded patient information. Following this the nine doctors discussed each case together and where possible agreed on a diagnosis (referred to as panel diagnosis 'PA'). We then ran these same 63 cases on our system.⁹

There was considerable disagreement between doctors on the diagnosis for a significant number of patients. On only 18 of the 63 patients did all 10 doctors (GP + 9 specialists) agree on the diagnosis. Our system also agreed with the doctors for all these patients.

There were an additional 22 patients where the specialist group after discussion amongst themselves came to the same diagnosis as the general practitioner. Of these 22 patients our system obtained the same diagnosis as the GP and panel for 20 cases. For the various specialists the comparable figure ranged from 9-20. It is worth noting that on two of the cases where our system failed to obtain the same result as the panel and the GP, none of the specialists obtained this result either. In three of the 20 cases where the system, the GP and the panel were in agreement, the system also suggested an alternative diagnosis suggested by 2-3 of the specialists.

This study seems to indicate that at least for those cases where the medical community itself can agree on a diagnosis, our system performs well. We note that only about half of these cases were so clear that all doctors reached the same initial diagnosis. The remaining half required discussion amongst the panel before agreement was reached. Even on such cases our system performed well.

In order to evaluate the effectiveness of the algorithm we logged the runtime behaviour of the system. The algorithm led to a quick focussing (within 3 loops) on the right direction to pursue in the taxonomy. Only in cases where no diagnosis was found did the system wander all over the taxonomy.

6 Discussion and Conclusions

There are a number of interesting conclusions that can be drawn as a result of the work presented here. Most importantly the *DTL* language appears to be one useful

⁹The full results of the evaluation summarised here can be found in [Zhang, 1993].

variant of a terminological language with defaults. Although traditional expressivity is limited the language includes defaults, which are necessary in a number of applications including the one described here. A default subsumption relation is described which combines principled default reasoning with classification, and it is indicated how this can be used in the T-box¹⁰ component of a terminological reasoner in order to build the default taxonomy. The default subsumption is also used in an interactive algorithm for guiding A-box reasoning. This work can also be regarded as an application which uses default reasoning techniques in a real system. To date there has been a large amount of work on theoretical aspects of default reasoning, but extremely limited implementation or application of theories developed.

In addition to indicating that the suggested language is in fact a useful trade-off in different kinds of expressivity, this work suggests that the distinction between primitive and defined concepts, (which is well established in terminological logics) is not always appropriate when using the system in an application domain. This could be an interesting aspect for further study. We observed that in this particular application the structure of the knowledge base was such that almost all stated inheritance relationships between concepts were between core concepts (concepts were linked by strict links in a mixed link graph). At the same time most properties were defined for default, rather than core concepts. It should be investigated whether this is a common pattern and if so what ramifications it has for terminological systems.

Although the approximate default classification algorithm presented here was adequate for this application, it should clearly be developed further to give a more complete algorithm.

The results of this work are also interesting from the point of view of diagnostic models or architectures for diagnosis systems. While our approach is in some ways close to abductive diagnosis (e.g. [?]) the hierarchic nature of the knowledge base in our system is an important differentiating factor. The use of an inheritance hierarchy in a diagnosis system is not in itself new, and has been used with notable success in systems such as Internist [Pople, 1977]. However the level of detail in our system is much coarser, and therefore much more readily available without large amounts of expert input. Internist relies heavily on probabilities for every feature in a disease description in order to drive the diagnostic process, whereas we have a much simpler 3 point scale (necessary, typical, possible). The fact that the diagnostic engine relies heavily on general purpose reasoning mechanisms rather than tailored rules is also significant for generalisability and maintainability. Domain specific heuristics can undoubtedly give better performance in many cases, but it is an advantage if these can be added on top of a relatively stable base architecture. The good diagnostic results despite lack of any domain specific heuristics, indicates that the combination of default reasoning and classification gives a powerful general

¹⁰The terms "T-box" and "A-box" refer to the components of a terminological system which reason about the terminology and about individuals respectively.

purpose reasoning tool.

The model of diagnosis as an interactive process which includes a principled search for information is a more realistic model in many situations than one which assumes complete information regarding the problem description.

A terminological logic framework was shown to be appropriate for implementing this medical diagnosis application. However it would not have been possible without the inclusion of defaults in the representational formalism.

Acknowledgements

We thank the medical doctors Dr Anders Johnsson, Dr Martyna Nowosielska and Dr Marek Woynarowski for help with the diagnosis application and Dr Toomas Timpka for the patient protocols used in evaluation. The first author has had useful discussions with Bernhard Nebel on terminological reasoning, including issues surrounding the integration of defaults.

References

- [Baader and Hollunder, 1992] Franz Baader and Bernhard Hollunder. Embedding defaults into terminological knowledge representation formalisms. In Bernhard Nebel, Charles Rich, and William Swartout, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the 3rd International Conference*, pages 294-305, Cambridge, MA, 25-29 October 1992.
- [Brachman et al., 1991] Ronald J. Brachman, Deborah L. McGuinness, Peter F. Patel-Schneider, Lori Alperin-Resnick, and Alexander Borgida. Living with classic: When and how to use a KL-ONE-like language. In John F. Sowa, editor, *Principles of Semantic Networks*, chapter 14. Morgan Kaufmann, San Mateo, California, 1991.
- [Braunwald et al., 1981] E. Braunwald, K.J. Isselbacher, R.G. Petersdorf, J.D. Wilson, J.B. Martin, and A.S. Fauci, editors. *Harrison's Principles of Internal Medicine*. McGraw Hill, 1981.
- [Donini et al., 1991] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. Tractable concept languages. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, pages 458-465, Sydney, Australia, August 1991.
- [Doyle and Patil, 1991] Jon Doyle and Ramesh S. Patil. Two theses of knowledge representation: Language restrictions, taxonomic classification, and the utility of representation services. *Artificial Intelligence*, 48(3):261-298, April 1991.
- [Fahlman, 1979] Scott E. Fahlman. *A System for Representing and Using Real-World Knowledge*. MIT Press, Cambridge, MA, 1979.
- [Findel et al., 1981] A.J. Findel, B.L. Burgess, M.R. Baker, and C.M. Fantal. *Current medical information and terminology*. Chicago: American Medical Association, 1981.
- [Horty et al., 1987] John F. Horty, Richmond H. Thomason, and David S. Touretzky. A skeptical theory of inheritance in nonmonotonic semantic network. In *Proceedings of the 6th National Conference of the American Association for Artificial Intelligence*, Seattle, WA, August 1987.
- [Nebel, 1990] Bernhard Nebel. *Reasoning and Revision in Hybrid Representation Systems*, volume 422 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Berlin, Heidelberg, New York, 1990.
- [Padgham and Nebel, 1993] Lin Padgham and Bernhard Nebel. Combining classification and non-monotonic inheritance reasoning: A first step. In *Proceedings of the Seventh International Symposium on Methodologies for Intelligent Systems*, Trondheim, Norway, 15-18 August 1993.
- [Padgham, 1989] Lin Padgham. *Non-Monotonic Inheritance for an Object-Oriented Knowledge-Base*. PhD thesis no. 213, Linköping University, Sweden, 1989.
- [Padgham, 1992] Lin Padgham. *Defeasible inheritance: A lattice based approach*. *Computers and Mathematics with Applications*, 23(6-9):527-541, 1992.
- [Patel-Schneider et al., 1990] Peter F. Patel-Schneider, Bernd Owsnicki-Klewe, Alfred Kobsa, Nicola Guarino, Robert MacGregor, William S. Mark, Deborah McGuinness, Bernhard Nebel, Albrecht Schmiedel, and John Yen. Term subsumption languages in knowledge representation. *The AI Magazine*, 11(2):16-23, 1990.
- [Pople, 1977] Harry E. Pople. The formation of composite hypotheses in diagnostic problem solving an exercise in syntetic reasoning. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, Cambridge, MA, 1977.
- [Quantz and Royer, 1992] Joachim Quantz and Veronique Royer. A preference semantics for defaults in terminological logics. In Bernhard Nebel, Charles Rich, and William Swartout, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the 3rd International Conference*, pages 294-305, Cambridge, MA, 25-29 October 1992.
- [Timpka, 1989] Toomas Timpka. *Design of Computer-Based Decision Support for General Practitioners*. Medical dissertations no. 298, Linköping University, Sweden, 1989. ISBN 91-7870-283-6.
- [Wor, 1977] World health organization, Geneva. *Manual of the international statistical classification of diseases*, 1977.
- [Zhang, 1993] Tingting Zhang. *A Study in Diagnosis Using Classification and Defaults*. PhD thesis, Linköping University, Sweden, 1993.