u Ottawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

u Ottawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Hongyang Wen
AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.Sc. (Systems Science)
GRADE / DEGREE

Systems Science
FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

The Development of Ontological Model for Clinical Decision Support System : A Case Study of Triage of Pediatric Hip Pain in the Emergency Department

TITRE DE LA THÈSE / TITLE OF THESIS

Professor Wojtek Michalowski
DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Professor Kevin Brand

Professor Herna Viktor

Gary W. Slater
Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

The Development of Ontological Model for Clinical Decision Support System: A Case Study of Triage of Pediatric Hip Pain in the Emergency Department

By

HONGYANG WEN

System Science Program

School of Information Technology and Engineering

University of Ottawa

Ottawa, Ontario, Canada

Hwen035@uottawa.ca

September 18, 2006

Thesis Advisor

Dr. Wojtek Michalowski

School of Management

University of Ottawa

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-25840-8
Our file Notre référence
ISBN: 978-0-494-25840-8

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

© Hongyang Wen, Ottawa, Canada, 2006

Acknowledgements

I will forever be grateful for the sound advice and constant support of my advisor, Wojtek Michalowski, for his feedback and for keeping me focused in my research. I would also like to thank Dr. Szymon Wilk for the sound support of the construction of knowledge model and Dr. Ken Farion for the detailed explanation of the clinical domain and suggestions on the user interface design. Thanks, to David Weiss for the help with MET environment and to Kelly Ketner at the Peer Help Centre, University of Ottawa, for her efforts in English editing.

Table of Content

Acknowledgements 2
Table of Content 3
Abstract 5
1 Introduction 6
2 Research Statement 8
3 Background 8
  3.1 CDSS 8
  3.2 MET Environment 12
  3.3 Pediatric Hip Pain 13
    3.3.1 Hip Pain ED Triage 13
    3.3.2 Hip Pain Retrospective Chart Study 15
4 Research Methodology 18
  4.1 Ontology 19
  4.2 Ontology Driven Design for KBS 23
  4.3 How to Create an Ontology: Ontological Engineering 29
5 Case Study in HP Domain: ODD 32
  5.1 HP Knowledge Model 33
    5.1.1 Experimental Design 34
    5.1.2 Results of the Experiment 42
  5.2 Ontological Model for HP Domain in MET 47
6 Discussion 55
References 58
Appendices 65
  Appendix A – Protégé 2000 65
  Appendix B – Weka 67
  Appendix C – Most Common Incorrectly Classified Patient Records for X-Ray Consult and X-ray_BScan Evaluation 68
  Appendix D – Missing Values in Testing Dataset 70

Appendix E – Data Mining Results 72
Appendix F – Rules That Misclassified Records In X-Ray Consult and X-Ray/BScan Evaluation 74
Appendix G – User Interface of the MET-HP for a Desktop Windows-based Computer 79
Appendix H – User Interface of the MET-HP for Reporting HP Triage Suggestions . 80
Appendix I – Detail of the 'HipPain' Class Implemented in Protégé 2000........ 81
Appendix J – Sample Decision Rules from the HP Decision Model 82
Appendix K – Abbreviations Used 83

Abstract

Clinical Decision Support Systems (CDSS), for providing patient specific advice, can only be accepted in clinical practice if they can fit in a clinician workflow. This would require such a CDSS to have diversified support capabilities, to be mobile, and to have flexible functionality. Such a system can be designed and developed only in a modular fashion where the high level abstractions describe the logic among different system components. Ontology, which is a formal specification of shared conceptualization, can be used to create a high level abstraction. Such decoupling of abstract CDSS logic from low level implementation facilitates developing and adding new applications and increases the reusability of different system components. In this research it is argued that a developed CDSS, according to ontology driven design with the ontological model of a problem domain expanded by a clinical decision support requirements, allows the creation of a system that is aligned with clinical workflow. In this research the proposed approach is illustrated with the CDSS for triaging pediatric hip pain (HP) in the Emergency Department. This application (called MET-HP) is created within the MET (Mobile Emergency Triage) environment that implements the ontology driven design principles. MET-HP is a mobile CDSS that includes a decision model derived from the analysis of retrospective chart data and it facilitates early triage of a child using incomplete data.

Keywords: Ontology; ontological model; ontology driven design; ontological engineering; clinical decision support system; data mining, knowledge model; knowledge based system;

1 Introduction

Clinical Decision Support Systems (CDSS)[1] are computer programs that are designed to provide expert support for health professionals making clinical decisions. The goal of these systems is to help health professionals analyze patient data and make decisions regarding diagnosis, prevention, and treatment of health problems [1, 2]. There are many systems that fit the definition. Among them there is a specific type of CDSS that are systems providing patient specific advice. In order to make this type of CDSS accepted in a clinical practice, these systems have to be able to fit in the clinical workflow[2] and be able to provide reliable recommendation to the clinician while not replacing the clinician's decision. This requires such systems to have multiple building blocks including medical domain knowledge model, support functionalities, and user interface components. The development of such systems would thus involve the development and integration of different system components. While such systems turned to be large scale, the interaction among these different system components becomes very complex. Thus, linking these different components together becomes an issue. It is rather difficult to integrate these components without a high level abstraction describing the logic among these components. As a consequence, how to represent the high level abstraction becomes one of the major concerns in the research.

Ontology, which is a formal specification of shared conceptualization [3], provides a convenient methodology to develop the high level abstractions that represent the logic of different CDSS components. It promises to provide a common understanding of problem domain, which can be communicated across people and computers [4]. Ontology has been applied already in research fields, like knowledge management, intelligent information integration, e-commerce, cooperative information system, database integration [5, 6, 7, 8, 53]. In recent years, there's a growing recognition that the ontological principles and concepts can be applied to the development of Knowledge

[1] Please note that all the acronyms in the paper are listed in the Appendix K.
[2] Here the clinical workflow refers to the tasks, resources, and triggers associated with a clinical process.

Based Systems (KBS) which are computer based systems having a symbolic representation of the knowledge and characterized by a separation of this knowledge from inference mechanism [9]. Such a KBS normally contains components including domain knowledge, problem solving methods and application user interfaces. While developing such a system, we believe that the high level abstraction describing the logic among different system components should be developed first. We can use ontology to represent this high level abstraction. Such an ontological model separated from low level implementation can then facilitate building new applications and increase the reusability of different system components. This is the main concept of ontology driven design (ODD) [6] which is a methodology that relies on ontological engineering to develop high level abstract models of the KBS structure.

In this research, we argue that a knowledge-based CDSS providing patient specific advice developed according to the ODD, with the ontological model of a problem domain expanded by clinical decision support requirements allows for easy expansion of system's functionality. In other words, if a CDSS is developed according to the ODD, then the question is, if expanding the ontological model of a problem domain by a clinical decision support requirement will allow for re-use of system's components. We are answering this question by developing the pediatric hip pain application (MET-HP) using Mobile Emergency Triage (MET) [10, 11] environment that was created according to the ODD principles. In order to develop this application, we created and integrated the abstract ontological model of a clinical domain enhanced with Emergency Department (ED) triage support functionality. The goal of the research is to show that ODD with enhanced abstract model allows building CDSS as required. Thus, accomplishing the goal we need to create an ontological model for pediatric hip pain (HP) decision support in the ED and then to integrate this triage support functionality into MET environment.

This thesis is organized as follows. We start with the research statement and background material on CDSS. As part of the CDSS description, we describe MET environment as an example of KBS CDSS developed using ODD. This is followed by a description of HP domain, including description of prospectively collected data. Subsequently, we give the theoretical foundation of this research including ontology and ODD methodology. Finally, we describe a case study and conclude with a discussion.

2 Research Statement

In this research we will demonstrate that expanding the ontological model of a problem domain with clinical decision support requirement allows taking full advantage of the ODD. We will further focus the research by considering a HP as a problem domain; support of the triage in ED of a hospital as defining support requirements; and MET as a CDSS development environment. This research involves development of ontological model for CDSS on a basis of clinical domain attributes and triage support requirements, capturing clinical acumen associated with triaging patients admitted to the ED with hip pain from the retrospective chart study, and representing it as a clinical decision model that forms a support component of the MET-HP developed using the MET decision support environment.

3 Background

In this section we review the CDSS and introduce the MET environment that will later be used to develop the triage support application for the pediatric hip pain presentation, MET-HP.

3.1 CDSS

CDSS is a computer program that is designed to provide support for health professionals making clinical decisions [12]. It has been gradually introduced since the early 1970s [13] to assist the clinicians in applying new information to patient care through the analysis of patient-specific clinical attributes. In general the CDSS can be categorized as the following [11, 14]:

1. Systems for accessing information that the physician must interpret. Accordingly, such systems are categorized as 1) medical information retrieval systems for managing and extracting medical knowledge and assisting in identifying the most appropriate sources of evidence appropriate to a clinical question[15], and 2) Electronic Patient Record System for managing patient data [11, 12]. One of the examples of CDSSs for information retrieval is MEDLINEPlus [16], which is an online information retrieval system that developed in response to the need for consumer health information and to extend awareness of quality health information resources available on the Internet.

2. System for focusing the user's attention. Such systems are designed to remind the user of diagnoses or problems that might have been overlooked [17]. They can provide to the clinicians the alerts and warnings about actions that might require attention. DoseChecker [18] is one of the examples of such CDSSs. It is designed to assist the staff pharmacists at Barnes and Jewish Hospitals with monitoring active drug orders for a set of drugs that must be carefully dosed for patients with possible renal impairment.

3. Systems for providing patient-specific advice. They provide advice based on patient-specific clinical data including diagnostic decisions (what the underlying health condition of the patient is) and management decisions (what treatment path the patient should follow) [11]. Many of these systems are used to enhance diagnostic efforts and include computer-based programs such as Dxplain™ [19] that provide extensive differential diagnoses based on clinical information entered by the clinician. Some systems in this category suggest a single best explanation for a patient's symptoms such as Internist-1 [20], which was a diagnostic system developed for general internal medicine that was capable of recognizing 600 diseases and of nearly 4,500 interrelated disease manifestations. Other systems

can provide management decisions to clinicians such as MET-AP [21], which was used for the management of pediatric abdominal pain in the ED.

CDSS from the first two categories have been relatively well accepted and used in clinical practice [11]. This is not a case with the systems belonging to the third category. Many researches [1, 11, 13, 15, 17, 22] argue that for these systems to be accepted, they need to satisfy three requirements:

1. They must fit in the patient management process, which means that CDSSs should provide decision support as part of clinician workflow. Using CDSSs that do not fit naturally in the management process would require additional effort from already busy clinicians. Also, fitting in the patient management process means the advice offered by the CDSSs should be at the point-of-care. Results in [22] show that systems that provided decision support at the time and location of decision making were substantially more likely to succeed than systems that did not provide advice at the point of care.

2. They should integrate with the Electronic Patient Record System (EPRS) and other hospital information systems so they can get patient information directly through EPRS and save users from re-entering available information [15].

3. They should offer reliable and comprehensive advice, which entails development of a model of both the required problem solving behavior and the clinical domain [17]. This also implies that such systems not only have to exhibit acceptable reliability, but also need to be able to explain and justify their recommendations[23].

While designing patient-specific CDSSs so it conforms to the above requirements, we have to transform these high[3] level requirements into specific design specifications. The transformation of these requirements means that:

1. Regarding availability at the point of care, such a CDSS has to be able to run on different platforms and the user interface should support easy switching between the platforms and the clinical presentations (different applications).

2. Regarding the integration with the EPRS, such CDSS should support common data communication protocol (for example, HL7) [11].

3. Regarding offering reliable and comprehensive advice, such a CDSS needs to have decision models that contain the proven knowledge of clinical domain. Moreover, the advice provided should be justifiable by providing the traceable path that such a CDSS used while arriving at an advice.

Considering a complexity of the above requirements, a CDSS that satisfies them is required to have multiple components including the user interface components, clinical domain description, and the problem solving methods. While designing such a system, the issue is how to integrate these components together. Having the high level abstraction that models the logic of these components and interactions among them should help with the integration. MET environment was developed to satisfy these requirements and it was used to create new clinical application described here. Basic ideas behind MET are described in the next section. It is followed by a description of the basic concepts needed for describing ontology and an illustrative example.

---

[3] The terms 'high level' abstraction are corresponding to the 'low level' implementation in a sense that the requirements can be easily understood and the implementation is carried out for the satisfaction of the requirements. Most of the time, the implementation can not be understood directly without knowing the specific requirements.

3.2 MET Environment

MET [11] is a mobile clinical decision support environment that can be used for developing and executing applications supporting triage [4] of various acute ED presentations. MET belongs to patient-specific CDSS and provides a triage suggestion on the basis of collected clinical information [11]. It uses ontologies to represent domains of various acute presentations, triage support functionalities, and the applications. It is designed to be used by physicians to triage a patient at the point of care [11]. Regarding the general requirements for a patient-specific CDSS, MET environment satisfies the three requirements by:

1. Having the capability to run on multiple platforms using pre-defined interface design derived from the analysis of patients' triage by ED physicians.

2. Having the ability to exchange information with the EPRS using the HL7 protocol.

3. Having decision models that were developed from the retrospective chart studies and that allow for the explanation of a path taken while generating the triage recommendations.

Each application developed using MET environment includes models of clinical domain knowledge, triage support functionality and problem solving method, and application specific user interface. MET uses ontologies and relies on ODD to represent the logical relationships among these components. In order to implement a new application in MET environment, all these three components have to be created following the logic of the system described in MET ontological model (more detailed description of basic principles of ontological modeling is given in section 4.2).

---

[4] The term triage refers to the initial assessment of a patient based on a set of clinical signs, symptoms, and tests [21]. It is the first stage in the process of patient management.

One of the MET applications – triage of pediatric abdominal pain (MET-AP) [10, 11, 24] has been successfully validated during a clinical trial. This thesis expands earlier research on MET by demonstrating how a new application MET-HP (triage of pediatric hip pain presentation) can be developed. In the next section we give a brief description of the hip pain problem domain.

3.3 Pediatric Hip Pain

3.3.1 Hip Pain ED Triage

Pediatric hip pain (HP) is one of the most common presentations of musculoskeletal disease in children. HP is often felt down the leg, at or just above the knee. This is also called referred pain. Non-referred hip pain may be felt in the groin or the upper outer thigh. Pain that starts in the low back is often felt in the hip region. Because the hip joint is so deeply located, it can be difficult to identify the exact source of pain. Also it is difficult to diagnose because it may be caused by numerous underlying conditions that could be benign or potentially very serious [25]. Moreover, the inability of young children to clearly describe the location and nature of their pain contributes to the diagnostic challenge [26].

The most common conditions associated with pediatric HP are: Acute Transient Synovitis (ATS), Legg-Calve-Perthes Disorder (LCPD), Septic Arthritis (SA), and Slipped Capital Femoral Epiphysis (SCFE) [25, 27, 28]. ATS of the hip is a self-limited inflammatory condition. It is one of the most common cause of HP in children manifested by unilateral hip or groin pain or sometimes medial thigh or knee pain that develops rapidly and either prevents ambulation or causes a limp [25]. Most children, two thirds to three fourths, with ATS experience complete resolution of their pain within a few weeks of the onset of symptoms [25, 62]. Although ATS usually occurs only once, a second event may occur in 4% to 17% of children [62, 67]. Treatment of ATS may include applying heat and massage, or if the patient is uncomfortable, hospitalization for observation and traction may be recommended [66]. LCPD is a disorder of the femoral head characterized by ischemic necrosis, collapse, and subsequent repair. It occurs in approximately 10% of patients. Although it is a self-limiting disease, 50% of untreated patients develop disabling osteoarthritis by the six year old. If severe, it can lead to permanent damage to the hip joint and early arthritis. The pain caused by LCPD may be localized in the groin or in the region of the thigh or knee and normally is intermittent. Thus, children may not complain of HP but instead may have a persistent limp. They may have decreased range of motion of the hip, especially rotation. An x-ray of the hips will often pick up changes in the femoral head that signify Perthes disease. In this case, bone scan is able lead to the diagnosis [65]. The treatment of LCPD is focused on femoral head containment and requires positioning of the proximal femur in abduction which can be achieved non-operatively or operatively [25]. SA, also known as infectious arthritis, is a serious infection of the joints [71]. The clinical symptoms for SA are very similar to ATS [29, 61, 67]. Patients with SA usually present a single swollen joint with acute pain on active or passive movement [28]. The usual history of a patient who has SA upon initial examination includes rapid onset of HP progressing to systemic illness and refusal to use the extremity [27]. It is considered as a surgical emergency because of the damage it causes to bone and cartilage and it may potentially create septic shock which is a serious condition [28, 29, 71]. SCFE is a disorder of the growth and development of upper femur [25]. The two most common features of the presentation of SCFE are pain and altered gait [61]. At least one-third of patients experience pain for 3 months or more before the diagnosis of SCFE is made. The true causes of SCFE are still unknown, although there are many theories [65]. The pain occurs either deep in the groin or along the distal medial thigh and knee and is often not associated with any specific injury, although it may increase in intensity from some minor trauma prompting the parents to take the child for an evaluation. On physical examination, the child will have painful range of motion of the hip. Rotation of the hip is especially limited. When the hip is flexed (with the knee bent), it will tend automatically rotate outward. As soon as the diagnosis is made, immediate treatment of SCFE is mandatory, with surgery being often the preferred method of treatment.

The ED triage recommendation of the pediatric HP can only be provided by the senior ED physician. The triage function involves categorization of the patients into one of three triage categories [24]:

1) Discharge, which means the patient has a benign and resolving condition, and can be discharged to the care of family physician. Normally patients who have HP because of ATS belong to this triage category.

2) X-ray consult, which indicates that the patient may have a bone injury. If the result of an X-ray is abnormal, the orthopedic surgeon is consulted. Patients suspected of LCPD would be triaged to this category.

3) Laboratory/X-ray/Bone Scan evaluation, which indicates the patient potentially has a serious condition such as SA, that is typically difficult to establish or rule out and needs to have more aggressive diagnostic evaluation [26] including blood work, an x-ray, a bone scan nuclear medicine test showing the blood flow to the joint and bone. Patients who are suspected of having SA or SCFE would be triaged into this category that always involves orthopedic consult.

The issue is to support ED physician decision making with the above triage categorization, so effective management path can be established quickly.

3.3.2 Hip Pain Retrospective Chart Study

ED triage of pediatric HP involves identification of commonalities associated with each of the triage categories. The ED physicians (EP) do it on a basis of their clinical acumen. Thus, the retrospective chart study of patients presenting with a hip pain to the ED should provide enough data to search for these commonalities. Considering that triage involves in a sense categorization of each of the patients on a basis of historical data, physical examination, and test results, it is possible to view it as a classification problem in artificial intelligence. Therefore, different data mining algorithms can be applied to patients' data in order to establish patterns associated with each of the triage categories. Such an approach was used in this research in order to capture knowledge associated with triaging pediatric HP and to include it in the application to be developed with a help of MET environment. We will further refer to this application as MET-HP. Data used for developing MET-HP comes from the retrospective chart study conducted during summer 2003 in ED of Children's Hospital of Eastern Ontario (CHEO). The study involved 412 HP patients, each described by up to 24 attributes given in Table 1. These attributes include historical observations, physical examination results, and basic laboratory tests. Continuous attributes were discretized according to the intervals consulted with the EP. Patients included in the study satisfied inclusion/exclusion criteria specified by EP and did not include any trauma cases or chronic HP cases. The ED discharge diagnosis was further validated with hospital records.

	Code	Description	Values
1	SEX	Gender	male, female
2	TEMPERATURE	Temperature	lt38, ge38
3	HEART_RATE	Heart Rate	lt40, ge40_lt80, ge80_lt120, ge120
4	AGE	Age	lt6y ,ge6y
5	PREV_VISIT	Previous visit to the ED	yes, no
6	HX_TRAUMA	History of mild trauma within last 2 weeks	yes, no
7	COMPLAINT_SITE	Location of pain	hip, leg, back, other
8	HX_ILNESS	History of systemic ilness	yes, no
9	GAIT_REPORTED	Reported gait	normal, limp, not_wt_bear (unable to bear weight on the leg)
10	DURATION	Duration of symptoms	lt24h, ge24h_lt7d, ge7d

11	APPEARANCE	General condition of a patient	well_NAD (No apparent distress), distress_unwell
12	HIP_REST	Position of hip at rest	normal, flexed, ext_rot (external rotation), flexed_ext_rot, int_rot (internal rotation)
13	HIP_ROM	Range of Motion of hip	normal, decreased
14	HIP_INT_ROT	Internal rotation of hip	normal, decreased
15	HIP_FLEXION	Flexion of Hip	normal, decreased
16	GAIT_OBSERVED	Observed gait	normal, limp, not_wt_bear
17	PAIN_ROM_HIP	Pain with ROM of hip	yes, no
18	OTHER_PAIN_SITE	Location of other pain	no, back, pelvis, leg, other
19	PAIN_PALPATION	Pain with palpation of hip	normal, tender
20	SWELLING	Swelling erythema of hip	yes, no
21	CURRENT_ILNESS	Current systemic illness	yes, no
22	PREV_PROBLEMS	Previous hip problems	yes, no
23	WBC	White Blood Cell	lt6, ge6_lt12,ge12
24	ESR	Electric Sedimentation Rate	lt24, ge24_LT40, ge40

Table 1 Clinical Attributes for HP

Collected data describes a classification problem with 24 conditional attributes and 3 decision classes – each corresponding to a triage category. The distribution of patients between the classes is given in Table 2. We have split the original data set into training data set and testing data set. In practical terms, it is common to hold one-third of the data out for testing and use the remaining two-thirds for training. Because we have only 412 patient records and we want to use more data for training the model, we used about 80% of the data for training part and the rest as the testing part. The division of a data set into two parts was done automatically using the appropriate function of Weka system (See

Appendix B for explanation). Table 2 shows the result of the data separation with the data visualization of the training data set and the testing data set. As we can see from Table 2, the HP data is heavily imbalanced because very few patients presenting in ED at CHEO have serious HP conditions. Such distribution is typical for HP population.

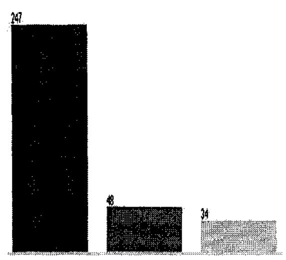
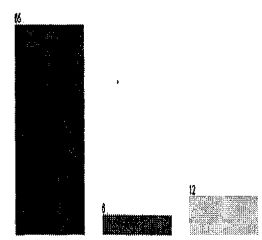
Label	Training Data Set		Testing Data Set	
DISCHARGE	247 (75%)		65 (78%)	
XRAY	48 (15%)		6 (7%)	
LAB_XRAY_BSCAN	34 (10%)		12 (15%)	

Table 2 Data Visualization[5]

4 Research Methodology

"Building knowledge-based systems today usually entails constructing new knowledge bases from scratch. It could be instead done by assembling reusable components. System developers would then only need to worry about creating the specialized knowledge and reasoners new to the specific task of their system. This new system would interoperate with existing systems, using them to perform some of its reasoning. In this way, declarative knowledge, problem-solving techniques and reasoning services would all be shared among systems. This approach would facilitate building bigger and better systems and cheaply..." [30]. Satisfaction of this reusability postulate asks for modeling of the interactions among the knowledge-based CDSS[6] components. For this reason, to obtain

---

[5] The columns represent the data that fall into corresponding triage categories following the sequence from left to right: Discharge, Xray, Lab_Xray_BScan.

[6] Please note that in our research, CDSS for providing patient specific advice satisfies the definition of KBS and thus is viewed as a KBS through out the paper. In this section we switch from knowledge-based CDSS

the high level abstraction describing the logic among different system components while separated from the low level system implementation becomes essential for the construction of such knowledge-based CDSS and ontology provides a convenient methodology to develop the high level abstractions. In this section we lay out the theoretical foundation of the research.

4.1 Ontology

The concept of ontology originated in philosophy, where it was used to represent a being or existence in a real world. Recently, it is used by the knowledge engineering community to represent knowledge about a problem domain [3, 5, 31, 32, 33]. There are many definitions of ontology from the knowledge engineering perspective. For our research, we want to use ontology to explicitly represent the abstract knowledge of clinical domain that may be reused and shared across applications and people. Thus, we adopt the definition by Studer [3] stating that "ontology is a formal explicit specification of a shared conceptualization", with an understanding that conceptualization refers to a specific problem domain.

Development of an ontological model usually involves specification of five main concepts [34, 35]. These are explained below using examples taken from the ontology developed for pediatric HP presentation. It is assumed that in this ontology a patient is described by a name, gender, birthday, and an episode of an illness.

- Class: a formal definition of individuals that share the same properties in a domain of discourse. Class can be concrete or abstract, existing or nonexistent. Abstract class is normally used only to organize the knowledge structure. Thus, no instance can be created for an abstract class. Taking the domain ontology shown in Figure 1 as an example, 'Patient' is a concrete class. Each patient has

---

to KBS considering ontology was first proposed in the knowledge engineering community and we will illustrate the theoretical foundation of this research from the knowledge engineering perspective.

certain properties such as first name, last name, age etc. Thus, the class 'Patient' would have the properties including 'FirstName', 'LastName', 'BirthDate' etc.

- Slot: property of each class. The property of a class can include characteristic of the class. For example, for each patient class, slots include first name, last name, gender, etc. The slot can be also used to describe the relations between two classes depending on a specific domain. This will be illustrated later while describing Relation.

- Facet (or role restriction): restriction on slots. Facets can refer to type of a slot (i.e. slot 'lastname' is of the string type, while slot 'visit_completed' is of the Boolean type). The other type of restrictions may identify if a given slot should be identified with one or many classes (i.e. slot 'gender' is identified with only 'Patient' class, while slot 'applies_to' is identified with more than one class (i.e. 'DecisionRules', 'UserInterface', and TriageSupport' classes)).

- Instance: an individual occurrence of a certain class. Individual instances are the most specific concepts represented in a knowledge base. For example, Angela, who is a patient, is an instance of the class 'Patient'. Note that in this research we use ontology with instances to represent a knowledge base. Instances can be only created for a specific class. Abstract class can not have any direct instances as it is used only for organizing classes. Please refer to [34] for details.

- Relation: a logical association among classes, slots, and instances. From the semantic point of view, there are three types of relations as listed below [36, 37, 38]:

  - Kind-of (isa): represents the inherited relation between classes; If class A is 'kind-of' class B, then it means that class B is a superclass of class A, and class A is a subclass of class B; all the slots defined in superclass B are inherited by the subclass A. For example, a scrotal pain presentation is a kind of presentation and thus the 'ScrotalPain' class is a subclass of 'Presentation' class. Slots defined for 'Presentation' class are inherited by 'ScrotalPain' class automatically at the creation of 'ScrotalPain' class.

- Instance-of: represents the relations between class and the instances of that class.

- Attribute-of: represents the relations between the instances and the slots.

- Domain specific relation: This type of relation is very specific to a given problem domain. We can not enumerate them all because it is domain dependent. For example, a patient had a visit in ED. So there has to be a relation between the patient class and visit class. In this example, we can define the relation of "each episode will deal with a single patient" by creating the slot 'patient' for class "Episode" which can only have one instance of class "Patient". And of course a patient may have several visits to the hospital. To represent this relation, we can create a slot named 'complaints' for class 'Patient' which can have multiple instances of class 'Episode' and thus 'complaints' is a slot of 'Patient' class. Both of these two relations were specific to the clinical domain.

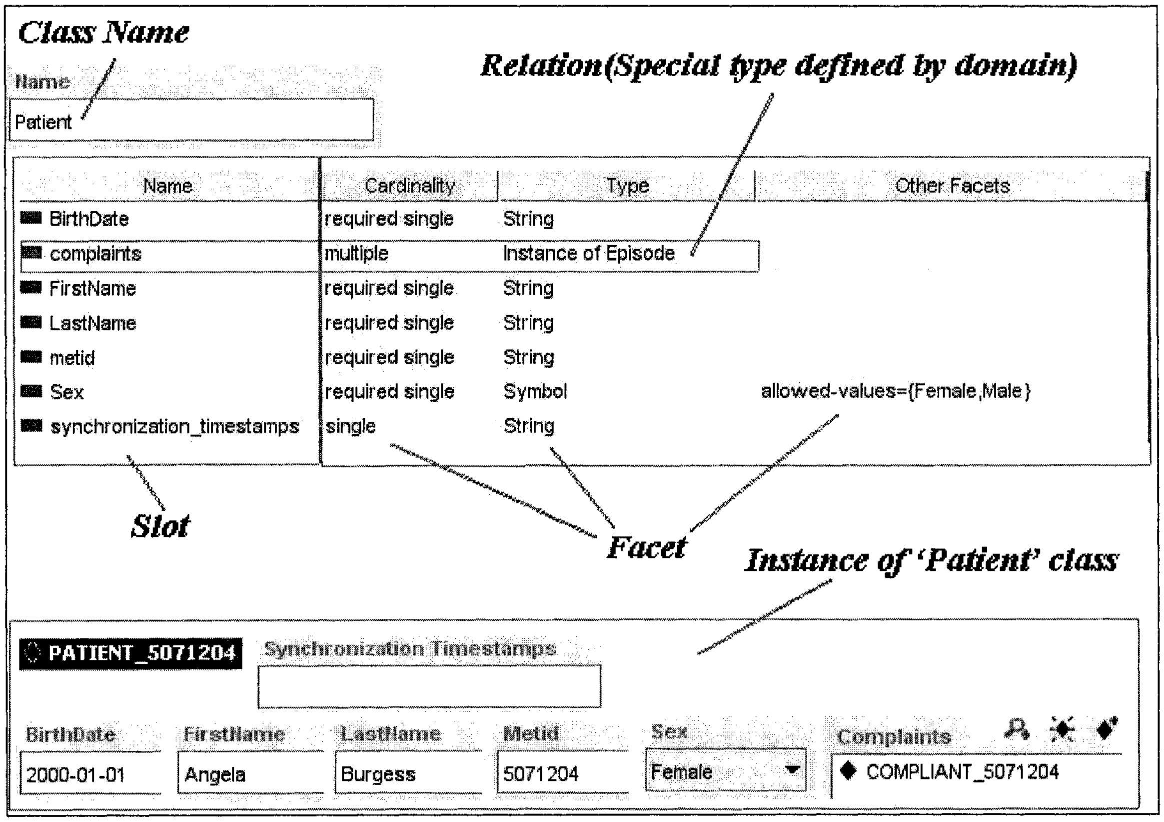


Figure 1 Ontology Illustration

Figure 1 takes all 5 elements described above together to show their interplay while describing the 'Patient' class. It is important to mention here that ontologies differ from controlled terminologies in that they represent the relevant concepts and relationships in a domain, whereas terminologies simply restrict the words used to describe the domain [39]. Ontology is used to model knowledge, thus, the classes represent main constructs on which such a knowledge is built, while the relations order these classes in a hierarchical structure [35]. In this research we view a knowledge base as the instantiation of the classes in the ontology (Classes and their instances). The reasons for using ontology in our research from the ODD perspective are:

1) Regarding the knowledge representation, there is a need for common understanding of the structure of information of a domain. This common understanding is provided by the high level abstraction of the system.

2) Regarding the knowledge structure, there is a need for writing the high level abstraction separated from low level implementation. This separation will enable the reuse of different components of CDSS and the reuse of the knowledge.

It has been argued in [40] that a KBS with a separate, explicit well-structured representation of the domain is simpler to develop, maintain and enhance than one in which domain terminology and relationships are embedded in the application code. This can be accomplished with the ODD of the KBS. This design paradigm and its application to the CDSS design are presented in the following subsection.

4.2 Ontology Driven Design for KBS

The ODD was initially proposed by Musen [6]. According to this design principle, one needs to represent on a logical level how individual components of the system are going to interact with each other and how to put all of the system components together. Such "knowledge" about system's operation needs to be represented in a formal manner and Musen proposed to use ontological engineering for this representation. The system's components normally include basic classes of a domain (domain ontology), abstract problem solving methods, and application specific description. Using ontology will help to write the high level abstraction which contained different classes representing the abstract descriptions of these components and the relations among these classes representing the logic of system architecture. This approach allows the separation of logical aspect of the system from the low level implementation. Because of this separation, it is possible to reuse individual components for building new applications [8]. In that sense ODD facilitates the system components reusability.

MET is a knowledge-based CDSS environment designed following ODD. The logic of different system components of MET was described in the ontological models. In the following, we introduce the existing ontological model in MET and illustrate the basic requirements for implementing a new application in MET in order to give the reader a sense how ODD can be applied to model the knowledge of logical operation of different system components in the CDSS.

The ontological model of MET environment is composed of three major parts described as domain ontology, support ontology, and application ontology [11]. Each part transforms a specific need into low level requirements. Figure 2 shows the existing ontological model of MET environment implemented in Protégé 2000 [41].

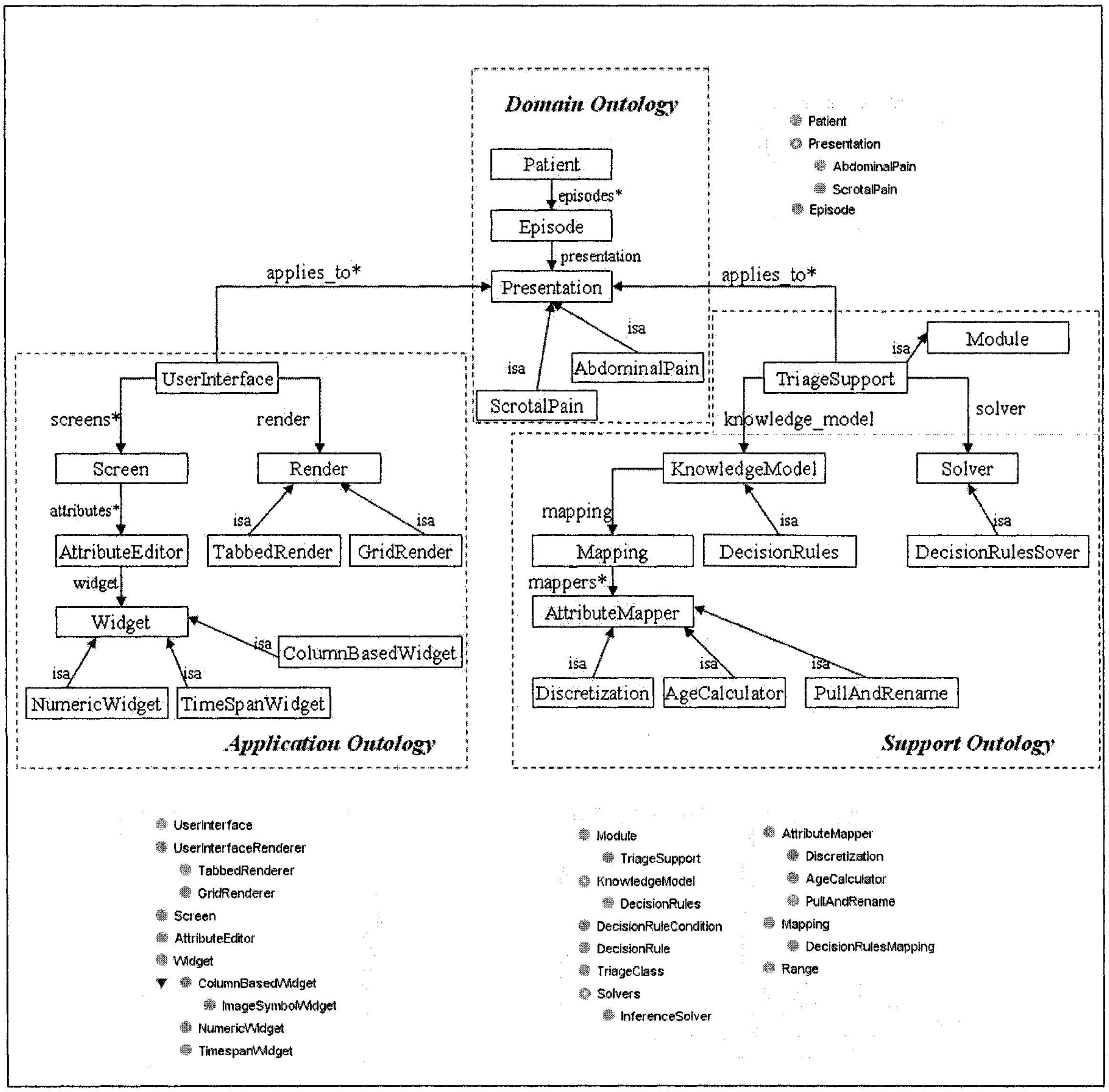


Figure 2 Ontological Model of MET and its Implementation in Protégé 2000

In the following, we will explain in detail what was modeled in each part of the ontological model and how these models need to be modified if new CDSS is going to be developed using MET environment.

- Domain Ontology – represents knowledge of clinical domain. It defines classes (patient, episode, presentation) and relations among them in the clinical domain. Currently MET includes a domain ontology common for all emergent acute presentations. This ontology was created assuming that:

    - A single patient can have several episodes (e.g., visits in the ED or hospitalizations);

    - Each episode is bound to a single clinical presentation; 'Presentation' is an abstract class that specifies all the clinical condition attributes used by a knowledge model. Currently two presentations have been implemented in MET environment. One is the abdominal pain and the other one is the scrotal pain.

  If a new presentation, which refers to a new clinical problem domain, is going to be added into MET environment, one has to provide the detail description of the attributes to be used in the knowledge model in that presentation. For example, in order to implement the MET-AP in MET environment, all the attributes of abdominal pain knowledge model have been described as slots specified in 'AbdominalPain' class. The rest of the classes including 'Patient' and 'Episode' in domain ontology should only be instantiated because we do not need to change the slots or the facets of these slots of these classes.

- Support Ontology – knowledge model of support functionality. It includes:

    - Class 'Module' describes the information of file repositories on server side which store the resource of implementing class libraries etc. (Regarding to the content of knowledge, this should be moved to application ontology in future.)

- Class 'TriageSupport' provides support functionality for decision making by the association with the decision model and solver; 'TriageSupport' class is implemented to describe what decision model to use for a particular presentation and what problem solving method to use to solve that decision model.

- Class 'KnowledgeModel' represents the knowledge necessary to support decision making; each presentation has a set of attributes and it falls into a certain outcome class. In order to implement a presentation in MET, a knowledge model should be developed first in order to support triage decision making. The 'KnowledgeModel' is the abstract class representing the different knowledge models used by different applications in MET.

- Class 'AttributeMapper' pre-processes values of attributes according to the specifications of a decision model and transforms the data into runtime inputs for problem solver [33]. The knowledge model built for the specific presentation expects the data to be in a certain format. MET has three types of attribute mappers represented as three different classes: 'Discretization', 'AgeCalculator' and 'PullAndRename'. 'Discretization' class describes the information of how to discretize the conditional attributes (defined as slots of the presentation class) specified in the source data according to the specific rules defined in each presentation. For example, the facet of slot 'temperature' of the 'Patient' class is specified as a real number, however, for HP presentation, it has to be discretized. These discretization rules are described in 'Discretization' class. 'AgeCalculator' class defines the rules for calculating 'age' attribute of patients. 'PullAndReName' class is used to describe those condition attributes that do not require a specific data preprocessing.

- Class 'Mapping' was implemented to link the 'KnowledgeModel' class with 'AttributeMapper' class. In other words, 'Mapping' class contains the description of group of attribute mappers for a specific knowledge model.

- Class 'Solver' describes the properties of a solution algorithm (solver) that would be used to execute the model using actual data to arrive at a solution. All decision algorithms, which are integrated currently in MET, share the same solver (but it is possible to have different solvers for different decision models).

To add a new clinical presentation, all classes of support ontology should be instantiated. (This is because of the ODD we do not need to change the structure of the knowledge but only to follow the predefined logic described in the ontological model and provide specific instances of the ontology.) 'TriageSupport' should be instantiated because for HP presentation there should be a new triage support to describe the corresponding knowledge model for this presentation and the solver used to solve this knowledge model. And thus, we have to instantiate the subclasses of 'KnowledgeModel' class such as 'DecisionRules' class etc. to describe what exactly this model is and to define the necessary attribute mappers for the specific knowledge model. And if the solver for the knowledge model has been implemented already, we can reuse the existing solver. Otherwise we have to implement the solver and in the 'Solver' class we have to provide the description of the solver such as the name of the solver and the path to it etc.

- Application Ontology – knowledge model of the user interfaces for different clinical presentations.

  - Class 'UserInterface' describes different displays used for different presentations. It represents logically the device-specific (desktop computer, mobile device, etc.) user interface for presenting and modifying patient's data and describes what screen renderer (TabbedRenderer or GridRenderer) is used, and what screens should be logically grouped together[7] for the specific presentation.

- Class 'Screen' describes what attribute editors should be used for the specific screen. It has slots, including 'name', the name of that screen; and 'attributes' which describes what attribute editors should be used. For example, as per physician's way of organizing patients' information, screens for scrotal pain and abdominal pain have been divided into three groups: History, Physical Examination, and Tests. Because these three groups have same set of properties, thus, instead of defining them as classes, they are created as instances of class 'Screen'. If a new presentation is needed, 'Screen' class should be instantiated accordingly to group the corresponding attributes. The order of the attributes displayed on the screen should be agreed by the EPs.

- Attribute editors bind specific widgets (editing tools) to specific attributes. 'AttributeEditor' class is implemented to describe what widget should be used for a specific condition attribute for the presentation.

To implement a new presentation in MET, we can reuse all of the application ontology and we only need to provide specific instances of the classes in this ontology.

The ODD implemented in MET environment allows to separate logical CDSS design from its physical implementation and to provide functionality required from the point of care support. Separation of the domain, support, and application models provides for development of the CDSS where problem description is viewed independently from the support requirement, and both are independent from the physical implementation of the

[7] We refer to these screens as logical screens.

system on a specific computing platform and for a specific user. The high level ontological model describes interplay between all these components and defines what needs to be added, modified, or re-used if new functionality, new type of users or new computing platform are to be included.

4.3 How to Create an Ontology: Ontological Engineering

Ontological engineering is a part of knowledge engineering and it refers to the set of activities that concern the ontology development process, and the methodologies and tools for building and supporting ontologies [5]. There are many methodologies of ontological engineering. We choose CommonKADS [34] methodology because of its wide acceptance and also because it has formality for building and evaluating ontologies and several supporting tools.

The CommonKADS methodology, which is the product of Knowledge and Analysis Design Support projects, has been described in [34, 42]. It defines a process for developing an ontology, proposes a modeling framework for representing domain knowledge to perform certain tasks, and it pre-describes a set of activities for developing KBS in the overall context of a domain, its current processes, and its needs [42]. Figure 3 shows the ontology engineering process defined in CommonKADS methodology.

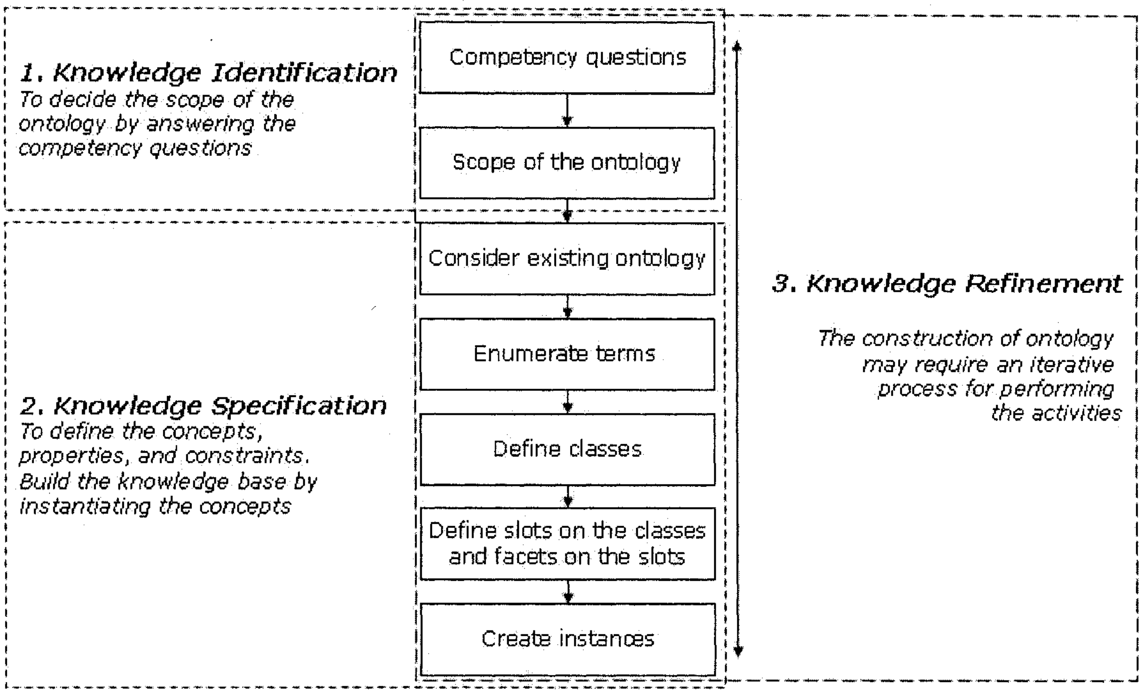


Figure 3 CommonKADS Methodology

According to [42], there are three knowledge-modeling activities: (1) knowledge identification, (2) knowledge specification, and (3) knowledge refinement. For performing these activities, CommonKADS proposes the following process to follow [34, 42, 43]:

Activity 1: Knowledge identification

1) Create a set of competency questions being a list of questions that should be answered using defined knowledge described in the ontology.

2) Clearly state the scope of the ontology as a function of the competency questions. Actually, at the outset of designing ontology, the competency questions are used explicitly to help with defining the scope of the ontology.

In this step, the ontology developers are expected to become sufficiently familiar with the domain so they are able to identify the domain concepts and potential reusable tasks.

Activity 2: Knowledge specification

1) Consider the existing ontologies for reuse.

2) Enumerate important terms in the ontology.

3) Define the classes in a hierarchical manner.

4) Define slots and facets.

5) Create instances.

Many basic rules have been created for these steps such as the naming considerations, new class or concept introduction, or ontology completeness.

Activity 3: Knowledge refinement

This provides an opportunity for the activities 1 and 2 to be iterated so that the user can modify the ontology according to the feedback loop. Depending on how familiar the ontology engineer is with the domain, knowledge refinement might have to be performed repeatedly.

Building ontologies is complex and time consuming process. It would require a significant amount of work if one has to develop the ontology by writing it in an ontology language directly, such as Web Ontology Language [44]. This is why the ontology development environments were created. According to [38], the following taxonomy of ontology tools exists:

- ontology development tools,
- ontology evaluation tools,
- ontology merge and alignment tools,
- ontology-based annotation tools,
- ontology querying tools and inference engines,
- ontology learning tools.

These tools provide supports in ontology editing, user interface, documentation, ontology import/export from/to different formats, etc [38, 41]. For the purpose of this research, we choose Protégé 2000 [41] as the ontology development tool. It is one of the most popular and advanced tools for ontology development. In Appendix A we gave a brief introduction to Protégé 2000.

5 Case Study in HP Domain: ODD

Lets illustrate potential use of the triage support CDSS that we are developing using simple scenario.

In the ED, the EP who was evaluating the abdominal pain patient using MET-AP application is summoned to see a patient who has just arrived, suffering from a severe HP. The EP has a mobile device with MET-AP client application installed. The EP has to be able to switch from MET-AP to MET-HP quickly. The EP has to make a triage decision whether the patient can be sent home to the care of a family physician, or he may have a bone injury Legg-Calve-Perthes Disorder and should have X-rays, or he may have a more serious condition and needs to have the clinical and laboratory evaluations plus more aggressive diagnostic evaluations. EP is going to use MET-HP to support her triage at the point of care. This application has to be pushed to EP's mobile device where it replaces currently residing MET-AP. Upon obtaining MET-HP, EP enters pertinent patient's data and invokes support functions to get triage recommendation.

From the above scenario we can see that there are three major issues while designing such application in an environment like MET. First, it has to fit in the EP's workflow so that the EP can switch from one application to another quickly. Second, it has to be integrated with the EPRS so that it can get the patient record quickly without wasting the EP's time from entering already existed patient data. Third, it has to able to provide reliable triage recommendation while not replacing but adding new information to be used by the EP. In order to provide such triage support on relatively lean computing devices that can be used at the point of care, a CDSS has to have modular design. Then the issue becomes how to integrate different system components together. In the following section we describe how we implemented MET-HP in MET environment.

5.1 HP Knowledge Model

In this section we describe development of the HP knowledge model. There are two approaches to derive knowledge: expert driven, whose aim is to obtain the knowledge models by acquiring it from experts, or discovery driven, which is the automatic discovery of knowledge from past data using appropriate knowledge discovery methodologies [45]. The development of the knowledge model, using expert driven approach turned out to be difficult because of the Feigenbaum bottleneck [46] associated with the verbal transfer of the domain expert mental skills. Because of the recent advances in applying discovery approach in clinical practice [17, 47] we chose the discovery driven approach to develop the knowledge model for MET-HP. As we have described in section 3.3.2, the problem of triaging HP patient is a classification problem for data mining. There are many classification methods (we will refer to them as classifiers) that can be applied to the construction of the HP knowledge model. In the following subsection we describe an experiment conducted to select the most appropriate classifier using an automatic knowledge discovery from retrospective chart data.

5.1.1 Experimental Design

In order to construct the decision model of HP domain, we need to choose classifiers that can satisfy certain criteria including:

1. Suitability for symbolic data with missing data values,

2. Model constructed by the classifier should be easy to present and comprehensible in a sense that the structure of such model should be easily understandable to not only modelers but also to the EPs.

Based on these criteria, we chose to evaluate the following classifiers using their implementations in Weka [48] system (Please refer to Appendix B for the description of Weka):

- Decision Tree (DT) – sometimes referred to as a classification tree, whose tree structure represents a classification model with nodes used for testing a particular attribute while its leaf as a classification that applies to all instances that reach the leaf, or a set of classification [48]. For example, a decision tree for HP domain should have the structure in which a node representing the patient conditional attribute, while each branch from that node represents a specific value of that attributes. The tree leaf represents one of the classification results. It would be Dicharged, or X-ray consult, or X-ray/Bscan evaluation in this case. To classify a HP patient's record, it follows the tree structure according to the values of the attributes tested in successive nodes, and when a leaf is reached the record is classified according to the class assigned to the leaf. The most advantage of DT over other classifiers is the ability to explain the results. Most people understand DT intuitively and this is why it was implemented by the CDSS [49, 50, 51]. In this research we used the J48 algorithm which is an improved version of the C4.5 algorithm [52] implemented in Weka. It uses divide-and-conquer approach to build pruned decision trees which are a widely used symbolic modeling technique for classification problems. When applying the DT to a new instance with missing values, it splits the instance into basic data items, evaluates part of it down each branch in proportion to the number of training instances going down that branch with numeric weighting method. When finally all these basic items reach the leaves, it combines the leaves by using weighting method [52].

- Decision Rules (DR) – relies on the logical statements in form of 'if [condition] then [decision] where condition is conjunction of elementary tests on certain condition attributes while the decision is an assignment to particular decision class [24]. For example, in the ED, if a male patient's gait is limp and he has no previous HP illness history with heart rate greater than 120 beats per minute, then he can be discharged home. This rule can be explicitly expressed as:

  if sex = male and gait = limp
  and previous_HP_illness = no
  and heart_rate > 120
  then triage = discharge

  Condition in this decision rule is 'a male patient's gait is limp and he has no previous HP illness history with heart rate greater than 120 beats per minute', while the decision is 'Discharge'. The DR, similarly to DT can be easily interpreted by a domain expert. Moreover, the model itself can be used for explaining and justifying a suggested triage [24]. There are many examples showing the success of applying DR to clinical domain [13, 17, 24]. In Weka, there are algorithms for the construction of DR, for example Part, which is based on J4.8 algorithm and builds a partial C4.5 decision tree in each iteration using separate-and-conquer technique and generate rules from the 'best' leaf. We chose not to use Part because the algorithm is based on J4.8 and the result would be expected similar to DT. We used LEM2 to construct the DR model. It has been already applied to develop the decision model for MET-AP [21]. LEM2 is based on rough-set approach and uses separate-and-conquer technique to build decision rules [21]. When applying a new instance with missing values, a similarity based classifier is used by assuming that the attribute with missing values can have any value possible. It finds the rules that can classify the instance and uses the strength of the rules to classify the instance.

- Naïve Bayesian (NB) – is a simple probabilistic classifier based on conditional probability models which assumes that the effect of an attribute value on a given class is independent of the values of other attributes. Learning in NB classification is same as estimating the joint probability distribution of the class and the attributes. It first calculates the conditional probabilities of each attribute, independently of the rest, given the particular class. Then it calculates the joint conditional probability of the class given the combined attributes based on the conditional probabilities of each attribute and the probability of that class. To classify a patient record, for example, we have to examine the conditional probability of the decision classes (e.g. for HP problem, the Discharge class, the X-ray consult class, and the X-ray/BScan evaluation class) given the particular attributes' values, and suggest the class that is most probable. The greatest advantage of NB is that classification can get the independent probability of each conditional attribute (e.g. the probability of a patient who has a temperature less then 38 degree Celsius should be discharged; the probability of a patient who is less then 6 years old should be discharged, etc) and then calculate the joint probability distribution of the decision class and the combined conditional attributes of the patient. For our research, we used NaiveBayes in Weka which is the implementation of the Naïve Bayes classifier. Each attribute is treated completely independent. When applying a new instance with missing values, NB simply calculates the probability based only on the attributes that are present.

In addition to these three classifiers, there are also other classifiers available such as logistic regression and neural networks. Logistic regression [56] is a statistical method that is used to estimate a probability that the outcome of the decision class assumes a certain value. This method is mostly used for handling numeric data and nominal attributes have to be first transformed into numeric attributes in order to be used for the construction of logistic regression model. We did not choose this classifier because it does not satisfy our first selection criterion. Also, the model itself can not be understood intuitively by the EPs. Neural networks [57] are analytic technique for information processing that is inspired by the way biological nervous systems, such as the brain, process information. The model itself is composed of a large number of highly interconnected processing elements called neurons working in unison to solve specific problems.. The model itself is like a black box. The network evaluates new data using its own internal model that is not explicitly stated and thus the classification result can not be explained directly to the EP. This does not meet our second criteria for choosing a classifier and we decided not to use this method.

In the construction of decision model, we will only consider traditional data mining classifiers which will generate a single model, for example, DT or NB, instead of using ensemble [59] learning methods which generate multiple models. This is because we want to build a decision model that can be easily understood by the EPs. This will not be a case with ensemble classifiers approach that while attempting to classify a new record, the ensemble passes it to each of its many models, obtains their predictions, and then combines them in some appropriate manner, for example, through voting for final class assignment. Specific examples of implementing the ensemble approach are bagging and boosting. Bagging [58] is a meta-algorithm that creates an ensemble of simple classifiers by dividing the learning set into several subsets and building a separate classifier from each subset, to construct a complex one. Boosting approach [58, 59] is to generate multiple classifiers and to derive weights to combine the predictions from those classifiers into a single predicted classification. The process of decision model construction using either bagging or boosting can not be understood intuitively by the EPs. Thus in our research we chose not to rely on ensemble approach.

Traditionally while evaluating the performance of a classifier, the overall accuracy of classification is always considered which is defined as a percentage of the ratio of the correctly classified instances to all classified instances [24]. This measure allows comparing and selecting classification methods and has been successfully used in several clinical applications, including the analysis of scrotal pain [24]. However, this measurement is often not sufficient and misclassifications from a more important to a less important class are treated exactly in the same way as misclassifications in the opposite direction [24, 63]. As clinical practice favors conservative behavior, where the latter mistakes are less more acceptable than the former ones, we decided to use some additional measurements together with overall accuracy for the classifiers' evaluation.

There are various additional measurements that can be used for the evaluation of the classifiers' performance. The great majority of these measures are defined for two-class problems. In our research we choose four additional measurements which are:

- F-measure that is used to measure a particular classifier's performance on the current class of interest,
- ROC provides a visual tool for examining the tradeoff between the ability of a classifier to correctly identify positive cases and the number of negative cases that are incorrectly classified,
- AUC is a measure of numerical accuracy of a model and it is used in conjunction with ROC,
- G-mean is a measure of balance between sensitivity and specificity.

In order to use these four measurements, we need to first dichotomize the original classes so that all the instances in the data set should be categorized into positive class and negative class [24]. In triaging HP patients, as the result of consulting with EP, both X-ray consult and X-ray/Bone Scan evaluations are important for the EP to identify because misclassification of patients in these two categories to Discharge category may potentially cause harm to the patients. Thus, we consider patients requiring X-ray consult or X-ray/Bone Scan evaluations belong to the positive class and those who can be discharged belong to the negative class.

Before we introduce the five measurements, we need to first define some terms that will be used for the calculation of these measurements. Table 3 shows the confusion matrix which represents the typical metrics for evaluating the performance of classifiers on binary class problems. In Table 3, the number of true positives (TP) and the number of false positives (FP) are calculated as TP/(FN + TP) and FP/(FP + TN).

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (the number of True Positives)	FN (the number of False Negatives)
Actual Negative	FP (the number of False Positives)	TN (the number of True Negatives)

Table 3 Confusion Matrix

F-measure [45, 48, 68] is the combination of Precision and Recall which are calculated as follows:

Precision = TP/(TP + FP)

Recall = TP/(TP + FN)

F-measure is defined as in the following:

F-measure = ((1+β^2) * Recall * Precision)/(β^2 * Recall + Precision)

where parameter β corresponds to the relative importance of precision versus the recall. Parameter β's default value is set to 1 which means that the precision and recall are equally weighted.

Receiver Operating Characteristics (ROC) curve [69] allows to distinguish between true positive rate and false positive rate as two separate performance measures. The higher the TP rate at low FP rate, the better is the model. ROC analysis has also widely been used in medical data analysis to study the effect of varying the threshold on the numerical outcome of a diagnostic test. A numerical measure of the accuracy of the model can be obtained from the area under the curve (AUC), where an area of 0.5 is equivalent to random guessing and an AUC of 1.0 suggests perfect accuracy [69, 70].

The Geometric Mean (G-mean) [64, 68] relates to a point on the ROC curve because it is calculated based on Positive Accuracy (also called Sensitivity) and Negative Accuracy (also called Specificity). Positive Accuracy is the probability that a positive patient is correctly classified [24]:

Positive Accuracy = TP/(TP + FN)

Negative Accuracy is the probability that a negative patient is correctly classified:

Negative Accuracy = TN/(FP + TN)

G-mean is defined as:

G-mean = ((Positive Accuracy) * (Negative Accuracy))^$^{1/2}$

The idea is to maximize the accuracy on each of the two classes while keeping these positive accuracy and negative accuracy balanced. For example, a high positive accuracy with a low negative accuracy will result in poor G-mean. Thus a classifier generating the highest G-mean is considered to have the best performance.

Considering that all of the HP conditional attributes are clinically important, we decided to use all of them in the experiment. The following procedure was designed to perform the experiment to construct the HP knowledge model based on so-called 'competitive evaluation of models', that is, applying different models to the same data set and then comparing their performance to choose the best:

1) For each simple classifier using all clinical attributes
   build a classifier from a learning set and verify it on a testing set
2) Compare different data mining methods and select the most appropriate classifier from classifiers we get from 1)

In the above procedure, we have mentioned the learning set and testing set as we have already introduced in section 3.3.2. The original HP data set was heavily imbalanced. This may hinder the performance of some standard classifiers such as DT or DR. Normally such problem can be handled through specific data mining techniques such as over-sampling or under-sampling [60]. Over-sampling refers to the process of increasing the number of records in the minority class while under-sampling refers to the process of decreasing the number of records in the majority class. Both of them involves changing the prior probabilities of the majority and minority class in the training set by changing the number of records in the majority and minority class. In this research we want to construct the decision model using original data set instead of artificially resample the training data set. Thus we did not choose these techniques for the training of a classifier. However we do want to point out that re-sampling of the training data set may possibly increase the performance of these classifiers for HP data set.

When comparing different data mining methods, we also need to check if the classification differences between the classifiers are statistically significant. We choose McNemar's test [54] which is a test on a 2x2 classification table as testing the difference between paired proportions where the hypothesis of interest is that the marginal changes in each of two independent samples' 2x2 table are equal. Suppose we have two classifiers, C1 and C2 (See Table 4 Example of McNemar's Test Result Table).

	Number of records correctly classified by C2	**Number of records incorrectly classified by C2**
Number of records correctly classified by C1	a	b
Number of records incorrectly classified by C1	c	d

Table 4 Example of McNemar's Test Result Table

Then the test result is $M = (|b-c|-1)^2/(b+c) > \chi^2$ with 1 degree of freedom. For a 95% confidence test, $\chi^2 = 3.84$. If M is larger than 3.84, then with 95% confidence, we can reject the null hypothesis that the two classifiers have the same marginal changes. If the result of McNemar's test shows the classification differences between the classifiers are not statistically significant, then we will choose the classifier based on the overall accuracy and the gain value.

5.1.2 Results of the Experiment

Table 5 presents the results of the performances of different classifiers on the testing data set.

	DT	DR	NB
Accuracy for Discharge	0.99	0.95	0.85
Accuracy for X-ray consult	0	0.17	0.33
Accuracy for X-ray/BScan Evaluation	0.33	0.58	0.58
Overall Accuracy	0.82	**0.84**	0.77
Positive Accuracy	0.22	0.56	0.61
Negative Accuracy	0.99	0.95	0.85
Precision	0.80	0.77	0.52
Recall	0.22	0.56	0.61
F-measure	0.35	**0.65**	0.56
G-mean	0.47	**0.73**	0.52

Table 5 Results of Different Classifiers on HP Testing Data Set

As we can see from Table 5, the performances of all the classifiers on the X-Ray consult category and the lab X-Ray/BScan evaluation category are very poor. The overall accuracies of all the classifiers are high possibly because of the highly imbalanced data set – the majority class Discharge includes 75.1% of all patient records in the training data set. DT misclassified all of the objects in X-ray category into the DISCHARGE category. NB gain slightly better results with the XRAY category comparing to LEM2 but lose the accuracy in the DISCHARGE category. All the classifiers had the worst performance on the X-Ray category comparing to the other two categories (For LEM2, only one out of six was correctly identified in X-Ray category and four of them were misclassified into the majority class DISCHARGE.). In order to identify the reasons of the misclassifications, we further examined the testing data set focusing on the X-Ray and X-Ray/BScan categories. We went though all the patients' records in the testing data set and listed all the misclassified data records by listing their indices and the details of the corresponding records in order to find some commonalities among them.

Details of the most common incorrectly classified patient records for X-ray and X-ray/BScan are presented in Appendix C. For the X-Ray consult category, we draw the following commonalities:

1. All misclassified patients have no previous problems (PREV_PROBLEMS) and with reported gait as limp (GAIT_REPORTED).

2. Three out of four patients were identified with complain site (COMPLAINT_SITE) as hip and the hip appearances (APPEARANCE) were all well with no apparent distress.

3. Those three attributes, including Hip at rest (HIP_REST), WBC, and ESR, have the most proportion of missing values.

For the X-Ray/BScan evaluation category:

1. All the patients have no previous problems.

2. No data for hip at rest (HIP_REST).

3. Four out of five patients do not have visual signs of significant inflammation (SWELLING) with one missing value for one patient.

4. Four out of five patients' hip appearances (APPEARANCE) were all well with no apparent distress (one exception with missing value).

5. None of the patients have a recent minor fall which may lead to a minor inflammation of the hip joint (HX_TRAUMA, three out of five values were NO and the other two were missing)

6. Comparing to X-Ray categories, there are no missing values in the WBC or ESR, which mean that these two attributes should be considered as the key attributes for determining whether the patient would need further lab evaluation or not.

Because it is much easier to conduct the analysis for potential reasons of misclassification using DR classifier, we further examined these most commonly misclassified records using LEM2 classifier to identify which rules exactly misclassified these individual records and how LEM2 misclassified these records. Please refer to Appendix F for details of the rules for classifying these record listed in Appendix C. We believe that there are three major reasons for such a poor performance in the two critical classes.

One of the reasons of the misclassifications in the two critical classes is mainly because the patients' data differ only on the missing values of clinical attributes. For example, record 3 was supposed to be classified to X-ray consult. According to LEM2 classifier, it was classified by rules 8 and 26 as following (Please refer to Appendix F for the explanation of the number following each of the rules named as the strength of the rule):

Rule 8:

(HEART_RATE = GE80_LT120) & (HX_TRAUMA = NO) & (GAIT_REPORTED = LIMP) & (DURATION = LT24H) & (PAIN_ROM_HIP = NO) & (PAIN_PALPATION = NORMAL)

⇨ (TRIAGE = DISCHARGE) (7.7)

Rule 26:

(DURATION = LT24H) & (APPEARANCE = DISTRESS_UNWELL) & (HIP_ROM = NORMAL) & (GAIT_OBSERVED = NORMAL) & (WBC = GE6_LT12)

⇨ (TRIAGE = DISCHARGE) (2.4)

In rule 8, record 3 satisfied all the conditions except value of attribute named as Appearance whose value was missing (Please refer Appendix D for the missing value in the testing dataset) and was treated the same as specified in the rule. Rule 26 works exactly the same as rule 8 where record 3 has missing value of attribute WBC and was treated the same as specified in rule 26. This reason can be applied to all the misclassified records in X-ray consult class and X-ray Bone scan evaluation class.

The second reason for such poor performance was that inconsistencies were indicated already by a classifier that provided ambiguous suggestions. In other words, more than one rule can be satisfied for a given patient data but pointing to different classifications. For example, record 58 matched to rules pointing at Discharge and X-ray consult:

Rule 16:

(HX_ILNESS = NO) & (DURATION = GE24H_LT7D) & (HIP_REST = NORMAL) & (GAIT_OBSERVED = NORMAL) & (PAIN_ROM_HIP = NO) & (OTHER_PAIN_SITE = NO)

⇨ (TRIAGE = DISCHARGE) (10.1)

Rule 18:

(SEX = MALE) & (HEART_RATE = GE120) & (GAIT_REPORTED = LIMP) & (PREV_PROBLEMS = NO)

⇨ (TRIAGE = DISCHARGE) (8.5)

Rule 75:

(SEX = MALE) & (AGE = LT6Y) & (HX_TRAUMA = NO) & (HX_ILNESS = NO) & (GAIT_REPORTED = LIMP) & (DURATION = GE24H_LT7D) & (CURRENT_ILNESS = YES) & (ESR = LT24)

⇨ (TRIAGE = DISCHARGE) (1.2), (TRIAGE = XRAY) (4.2)

According to LEM2 classifier, it will categorize records by adding the strength of the rules and give the recommendation by choosing the strongest rule (the highest value of the strength of the rules). For example, in the case for record 58, rule 75 can recognize the record should be categorized to X-ray consult with the value of strength of the rule as 4.2, higher than 1.2 for Discharge. However, rule 16 and rule 18 also apply to the record. So the strength of rules categorizing Discharge class would be adding all the strength of individual rules together. Because of uneven distribution of learning objects, even on a percentage ratio, the strength of rules for categorizing records to majority class Discharge would be still higher than the other two classes. Thus, this would lead to the final decision to Discharge class instead of X-ray consult class.

In order to improve the performance of the classifiers in X-Ray and X-ray/BScan evaluation classes, we also need a further study on the latent clinical attributes for identifying the patients who should be discharged home or may need clinical consult. Such latent attributes, however, can not be collected from the retrospective chart study.

The result of the experiment showed that DR achieved the highest values in overall accuracy, F-measure, and G-mean. To better understand the performance of the different classifiers, we also present the ROC analysis results of different classifiers on the testing data set. In Figure 4, we can see that the AUC of DT is lowest comparing to those of the other two classifiers. The AUC of DR is slightly higher than NB's by around 0.06. DR's TP rate is higher than NB's when FP rate is less than 0.4. When FP rate is higher than 0.4, NB's performance is slightly better than DR. However NB's G-mean value comparing to the G-mean measure of DR is lower which indicates that DR performs better in a sense that it has better result in the classification of positive cases while not erring too much on the negative cases.

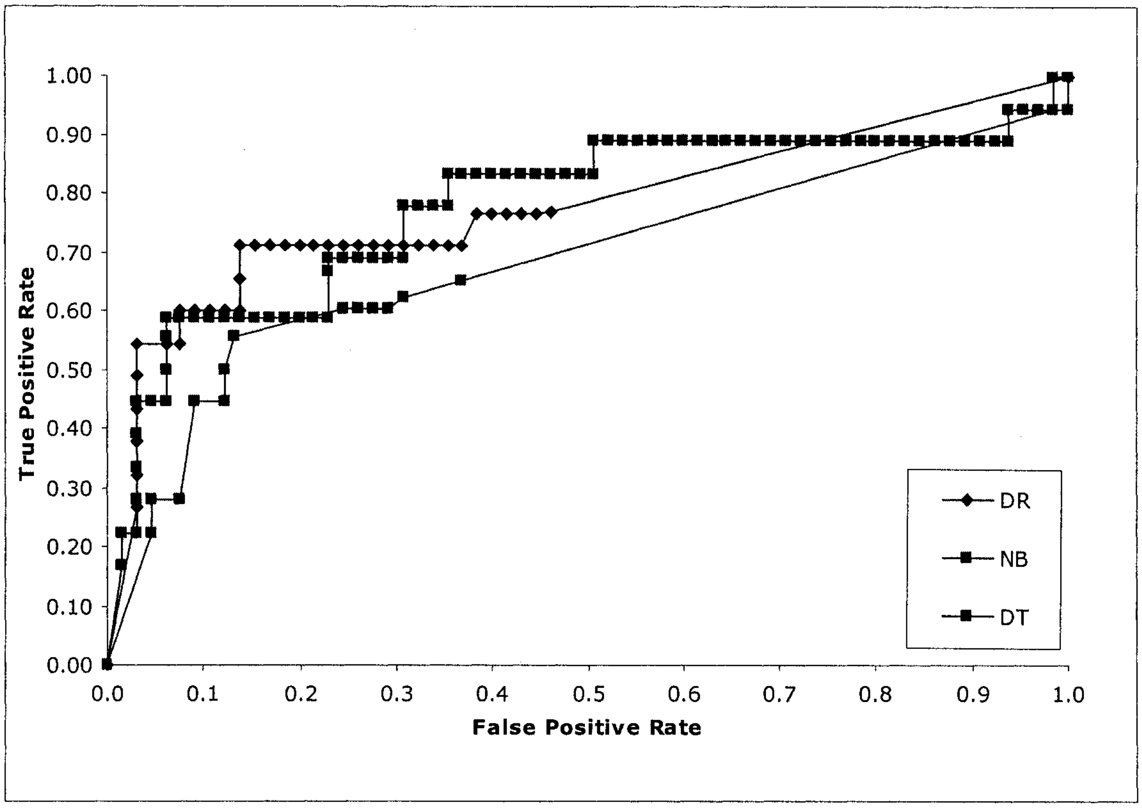


Figure 4 ROC Curve for the Classifiers

Table 6 shows that the results of McNemar test indicating that the differences between classification results for these classifiers are not statistically significant.

	NB	DR
DT	0.64	0.13
NB		2.08

Table 6 McNemar Test Result for Compared Classifiers

Thus we chose to implement in MET-HP the classifier developed by DR. Appendix J gave the sample decision rules in the HP knowledge model.

5.2 Ontological Model for HP Domain in MET

In this section we will describe the development of an ontological model for HP domain in MET environment using the CommonKADS [42] methodology. Based on this methodology, in the first stage as knowledge identification, we defined the scope of the ontology is about developing MET-HP in MET environment. We do not need to list the competency questions because the HP ontological model should be built based on the existing ontological model in MET environment. Thus, we can skip the first stage of the CommonKADS methodology and create the knowledge base by instantiating the corresponding necessary classes as we have illustrated already in section 4.2.

The existing ontological model in MET environment has three elements: 1) the domain ontology describing information of clinical domain such as patient, episode, and characteristics of HP etc.; 2) support ontology describing the knowledge of providing triage support for HP problem domain, classification of decision models, the problem solver for the a certain type of decision models, and the mapper to preprocess values of attributes according to the specifications of a decision model and transform the data into runtime inputs of problem solvers; 3) application ontology including editing modules for user interface.

In order to develop the ontological model of MET-HP, we have to able to expand the existing ontological model in MET environment by instantiating each part of the ontology accordingly, while satisfying the HP decision support requirements. In the following, we show how we developed the MET-HP application by expanding and re-using the ontological model in MET environment.

- Domain Ontology (See Figure 5 for the implementation of the domain ontology in Protégé 2000)
    - 'Patient' and 'Episode' classes are reused for MET-HP.
    - HP is a new presentation in MET and thus, we have to create a new class named as 'HipPain' with slots associated with the clinical attributes given

in Table 1. Appendix I shows the detail of 'HipPain' class implementation in Protégé 2000.

Now all the classes needed in domain ontology have been presented. We want to provide an example to illustrate how these classes can be used for HP domain. A patient named 'John Morgan' visited the ED for his HP. The information of the patient such as name and gender is described in the instance of class 'Patient' named as 'met-db_instance_110001'; the information of the episode is described in instance of class 'Episode' named as 'met-db_instance_110002'; and the detail of his clinical conditions is described in instance of class 'HipPain' named as 'met-db_instance_110000'. (The rest of the explanations will be the similar to this example, thus, we will not enumerate all of those instances created under each classes in the following examples. However readers can get the information from the corresponding figure by tracing the black line drawn in the picture.) Figure 5 shows this example of the instances created under each of the classes in the domain ontology.

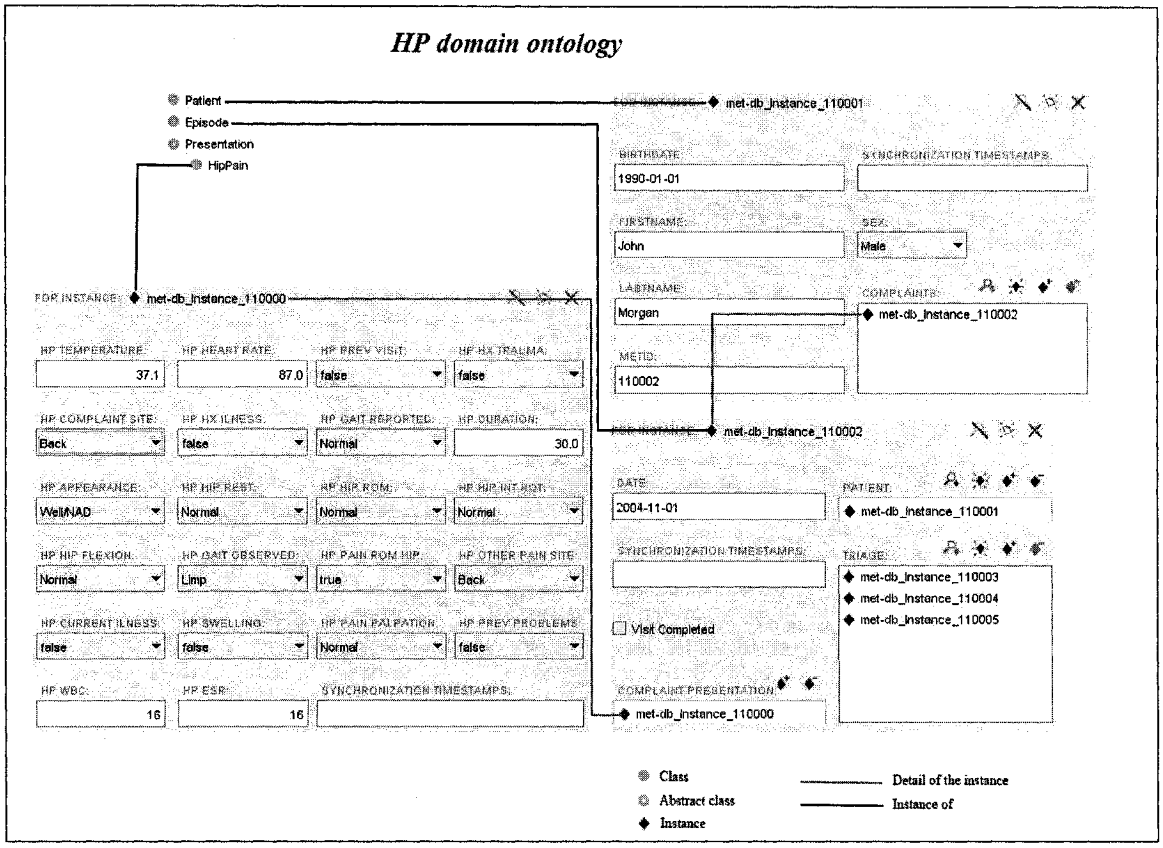


Figure 5 HP Domain Ontology Implemented in Protégé 2000

- Support Ontology (See Figure 6 for implementation of the support ontology in Protégé 2000 and Appendix H for the screen shot of the HP triage suggestions) – Since the DR is a decision model for HP, the existing solver can be re-used.

    - o Decision model: HP decision rules model 'Rule_Model_HP' is an instance instantiated from class 'DecisionRules', which describes what problem domain the model should be applied to (class named as 'HipPain' in this case); what mapper should be used (instance named as 'MAPPINGS_HP'); what rules have been specified in the model (instances named as 'HP_RULE_1', 'HP_RULE_2' etc.); each instance instantiated from class 'DecisionRule' (See Figure 2), for example 'HP_RULE_1', are described with two slots: 'rule_conditions' and 'triage_class_suggestion'. The triage class suggestions for HP fall into three categories (See Appendix H). Thus, the facet of the slot 'triage_class_suggestion' specifies that only one of three values representing the three categories can be applied to a single instance of 'DecisionRule' class. Additionally, the strength factors are calculated for the recommendations pointing to each class [24]. Each recommendation is presented as a relative value depending on the calculated strength of rules (please refer to Appendix F for the relative strength of a rule). Appendix H is a screen shot of the HP triage suggestions.

- Solver: re-use 'InferenceSolver'.

- Attribute mappers: we have three types of attribute mappers as described in MET section. In order for the solver to solve (make inferences) the HP decision model, we have to instantiate the three classes according to the discretization rules defined in HP presentation. We reused the instance of 'AgeCalculator', which was a class describing the method implemented for calculating the age of a patient in the format of 'YYYY-MM-DD'. According to the discretization rules defined by the EPs, we have six clinical conditional attributes to be discretized, including temperature, heart rate, age, length of symptoms, WBC, and ESR. We can directly use data for the rest of the attributes. (See Figure 6 green line part)

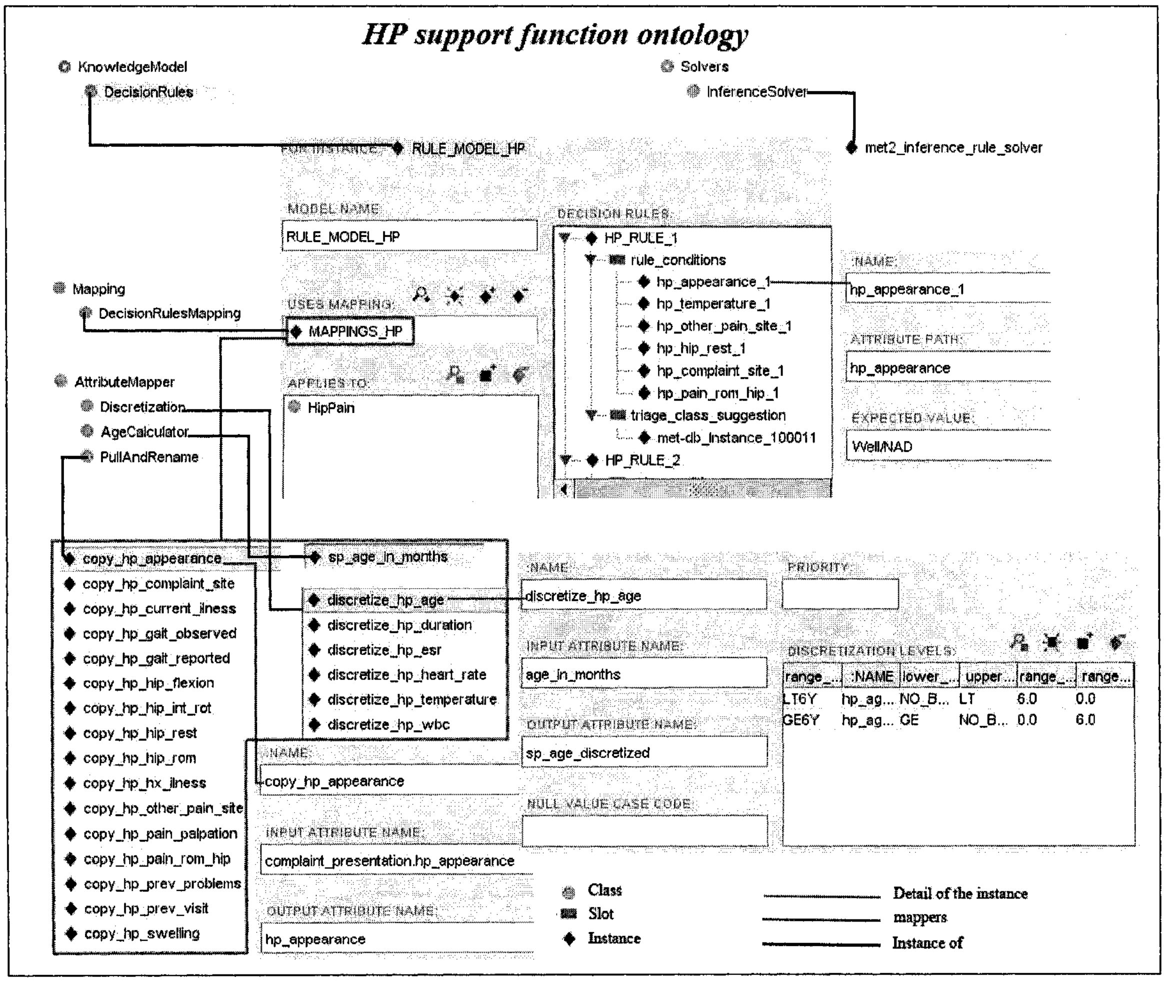


Figure 6 HP Support Function Ontology Implemented in Protege 2000

- Application Ontology (See Figure 7 for implementation of the application ontology in Protégé 2000 and Appendix F for the screen shot of the HP presentation for a PC computer) – now we need to describe the knowledge of application specification for HP presentation.

    - We have created two user interfaces as the instances of class 'UserInterface': 'hp_generic' for generic user interface and 'hp_pc-default' for pc user interface. We do not need to know how exactly these user interfaces were implemented in the application, but we do need to know what these instances represent logically so that we know how to manage the different system components. The difference between these two instances is that 'hp_generic' is used to represent the display on mobile devices and thus, requires using tab instead of grid display that requires more screen space. User interfaces specifications need to associate devices to be used (pc-default or generic?), screen render to be used, and screen groupings to be used for specific presentation. We use 'hp_pc-default' as an example to show what the user interface instance created for HP presentation would be. In this case, it is displayed on a workstation (specified as 'pc-default') and uses a specific grid renderer 'render_grid_2_columns'. Three screens are grouped together to display the HP patient data (See Figure 6 for the detail of the 'hp_pc-default' instance).

- o Three instances of class 'Screen' are created: 'hp_screen_history', 'hp_screen_phys_exams', and 'hp_screen_tests'. Each of these instances represent the logical screens that group and manage several HP attributes as per EPs' suggestions. These instances are taken as the values of slot 'screens' of the instance 'UserInterface' named as 'hp_pc-default' representing the screen used for displaying the patient's conditional attributes on PC.

- o As we have described previously, each instances of class 'Screen' manage several different HP conditional attributes. For each attributes, there are different components (called widget) used for their editing. We have to create instances of classes including 'ColumnBasedWidget', 'NumericWidget', and 'TimeSpanWidget' accordingly to manage different widgets for editing different attributes. For example, an instance of class 'ColumnBasedWidget' named 'enum_inline_chboxes_1_column' is used for the instance of class 'AttributeEditor' named as 'editor_hp_appearance'. This instance ('enum_inline_chboxes_1_column') was reused to inform the scrotal pain ontology. Sometimes a specific attribute needs to have some unit measure to be displayed on a widget which is very specific for the presentation. We have to create an instance of class 'Widget' or its subclasses depending on what widget the specific attribute is going to use as an editing component. For example, the HP attribute heart rate which is a numeric type needs to have a unit measure 'bpm' (beats per minute) to be displayed on the widget, thus we created an instance of class 'NumericWidget' with unit name specified as 'bpm'. (See the left bottom part of Figure 7).

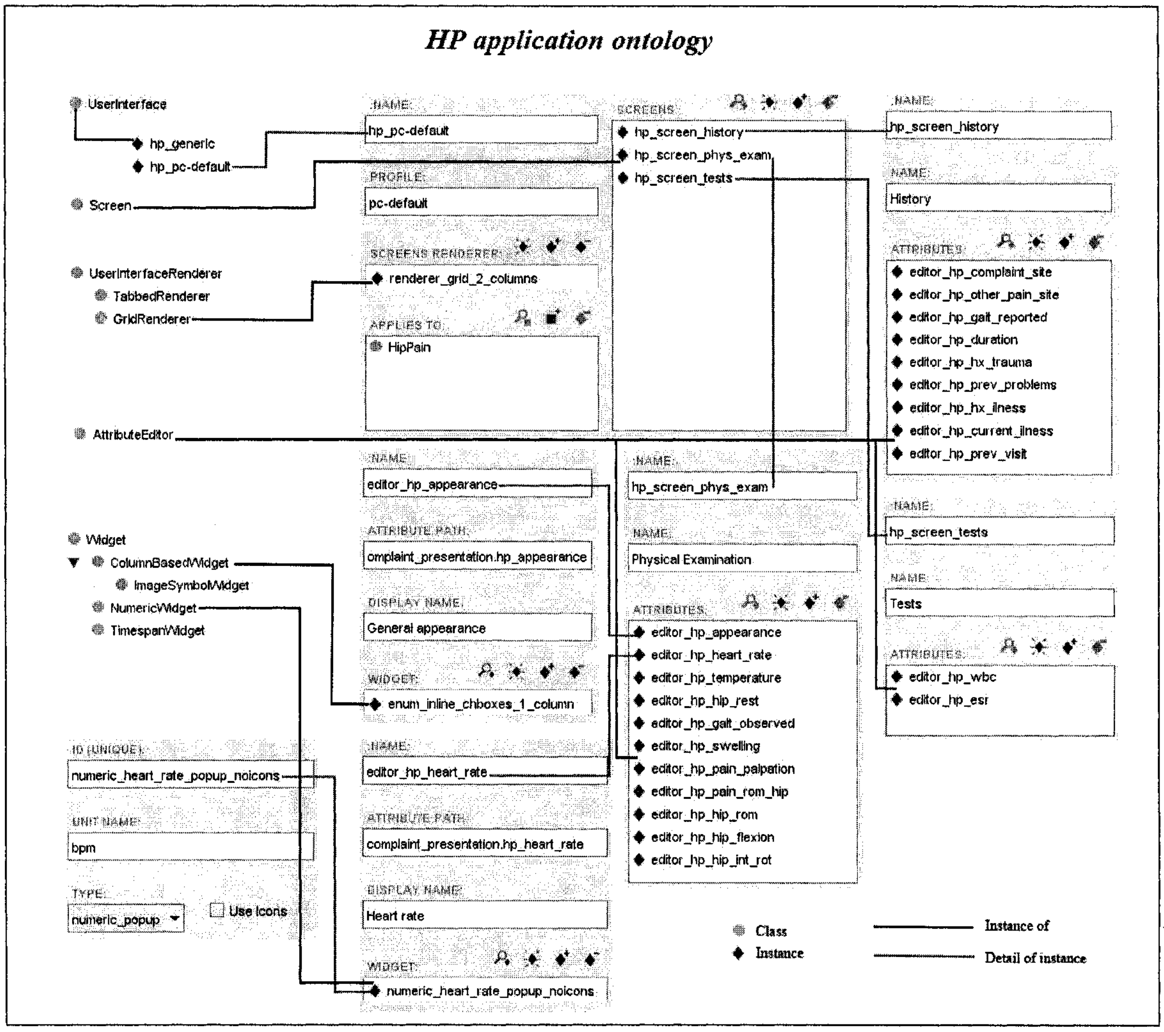


Figure 7 HP Application Ontology Implemented in Protégé 2000

We presented development of the ontological model for HP domain by expanding the existing ontological model in MET environment. While building MET-HP, we reused most of the classes in the existing ontological model without modification of the system logic. The structure of knowledge to be added can be easily identified. This means that we need to follow the pre-defined system logic while implementing an application in a knowledge based CDSS environment. We do need to know logically what each part of the ontological model represents without being concerned about the detail of the actual low level implementation of the specific application. In that sense, we can say that ODD facilitates the implementation of CDSS and the reuse of existing system components.

6 Discussion

While designing CDSS for providing patient specific advice, such a system has to integrate the multiple system components in order to comply with a requirement of providing support at the point of care. We believe that a CDSS based on ODD in which the high level abstraction describes the logic among the system components can facilitate new application development and increase the reusability of different system components. In order to validate such a system design, we did a case study of triage of pediatric HP in the ED and implemented it as MET-HP application in MET environment. This involved the development of HP knowledge model, the development of HP ontological model based on existing ontological model in MET, and the integration of this new triage support functionality into MET environment. The case study has shown that a CDSS, which was developed according to ODD, with the ontological model of pediatric HP domain expanded by clinical decision support requirements, allows for relatively easy creation of new clinical application.

The high level abstraction describing the logic among different MET components has been predefined in the existing ontological model. Because of the ODD, we could easily expand this model by instantiating the corresponding concepts and providing new necessary elements as specified in the ontological model of a MET environment. The straightforwardness of creating the MET-HP application has proven adding new application into MET environment does not require change of the logic of the system when a pre-defined logic described in the ontological model is followed. Thus, we can state that ODD facilitates the expansion of system's functionality by allowing for seamless addition of new applications supporting new ED presentations.

We have also observed that ODD is suitable for designing a small mobile system in which one logical model can describe the logical operation of different system components. If such a system was designed differently, then the logical operation of the system components could not be described independently from the implementation. As a result, if a new clinical presentation should be supported, then such an application had to be implemented independently. In that sense, it would be impossible to reuse existing system components and the entire application should be developed from the beginning resulting in duplication of common elements.

In this research we have also noted that ontological engineering facilitates ODD of knowledge based CDSS. We have partially applied CommonKADS, to construct the ontological model of HP domain. We did not have to rely fully on the CommonKADS because the HP ontological model was build re-using parts of the existing ontological model in MET environment. However, even such limited application of knowledge engineering tools show that they can help with the construction of the knowledge model at the early stage of the development of knowledge-based CDSS, and thus, can facilitate the development of CDSS.

While constructing the decision model for HP domain, we found that DR classifier operates relatively well on HP data. However, we did observe its poor performance for the two critical classes, namely X-ray consult and Laboratory/X-ray/Bone Scan evaluation. Other classification methodologies should be explored to better reflect conservative decision-making of EPs. Considering that focus of our research was to validate usability of the ODD using MET environment, we did not pursue this avenue of research further as being beyond the scope of this work.

The HP case study and the resulting MET-HP application should be considered as a pilot study for the validation of the ODD for specific clinical application and specific decision support setting. This is further exemplified by the results of testing of the HP decision model that show that such a model (even if implemented in well-designed application) is far from being ready for clinical evaluation. However, if accurate and comprehensive model is developed and further evaluated for its compliance with medical knowledge, then MET-HP should be ready for prospective testing in clinical practice.

References

1. Mendonca E.A.. *Clinical Decision Support Systems: Perspectives in Dentistry.* Journal of Dental Education. 2004. 68(6). p. 589-597.
2. Soumeya L. Achour, M., Claire Rieux, Philippe Bierling, and Lepage E.. *A UMLS-based Knowledge Acquisition Tool for Rule-based Clinical Decision Support System Development.* Journal of American Medical Informatics Association. 2001. 8(4). p. 351- 360.
3. Studer R. and Fensel D.. *Knowledge Engineering: Principles and Methods.* IEEE Transactions on Data and Knowledge Engineering. 1998. 25(1-2). p. 161-197.
4. Su X. and Llebrekke L.. *A Comparative Study of Ontology Languages and Tools.* Proceeding of Conference on Advanced Information System Engineering. 2002. Toronto, Canada. p. 761-765.
5. Asuncion G.P.. *Ontological Engineering with Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web, 3rd Printing.* 2004. Springer.
6. Musen M.A.. *Ontology-Oriented Design and Programming.* Knowledge Engineering and Agent Technology. 2004. IOS Press. p. 3-18.
7. Kalinichenko L.A., Missikoff M., Schiappelli F., and Skvortsov N.. *Ontological Modeling.* Proceeding of the 5th Russian Conference on Digital Libraries. 2003. St.-Petersburg, Russia. p. 142-149.
8. Kishore R., Zhang H., and Ramesh R.. *A Helix-Spindle for Ontological Engineering.* Communications of the ACM. 2004. 47(2). p. 69-75.
9. *CEN/TC 251 European Standardization of Health Informatics.* [Available from: http://www.centc251.org/.
10. Michalowski W. and Wilk S.. *MET System: A New Approach to M-health in Emergency Triage.* Journal on Information Technology in Healthcare. 2004. 2(4). p. 237-249.

11. Michalowski W. and Wilk S.. *Design and Development of a Mobile System for Supporting Emergency Triage.* Methods of Information in Medicine. 2005. 44 (1). p. 14 - 24.

12. Musen M.A. and Shortliffe E.H.. *Clinical Decision Support Systems*, in *Medical Informatics.* Computer Applications in Health Care and Biomedicine, 2nd Edition. 2001. Springer-Verlag. p. 573-609.

13. Kulikowski C.A.. *Artificial Intelligence in Medicine: a Personal Retrospective on Its Emergence and Early Function.* Proceedings of ACM Conference on History of Medical Informatics. 1987. Maryland, U.S.A.. p. 199.

14. Shortliffe E.H.. *Computer Programs to Suppport Clinical Decision Making.*. Journal of the American Medical Association. 1987. 258. p. 6 1-66.

15. Coiera E.. *Clinical Decision Support Systems.* The Guide to Health Informatics, 2nd Edition. 2003. Arnold Publishers.

16. Bronson R. and Fitzpatrick M.. *What Every Medical Librarian Should Know About MEDLINEplus.* Medical Reference Services Quarterly. 2001. 18(4). p. 11-17.

17. Musen M. and Shortliffe E.. *Clinical Decision Support Systems.* Medical Informatics. Computer Applications in Health Care and Biomedicine, 2nd Edition. 2001. Springer-Verlag. p. 573-609.

18. Kahn M.G. and Marrs K.A.. *Creating Temporal Abstractions in Three Clinical Information Systems.* Proceedings of the 19th Annual Symposium on Computer Applications in Medical Care. 1995. L.A., U.S.A.. p. 392-396.

19. Robert T.. *Clinical Decision Support System.* Making Health Care Safer: A Critical Analysis Patient Safety Practices. 2001. p. 589.

20. Miller R., McNeil M., Challinor S., Masarie F., and Myers J.. *The INTERNIST-1/Quick Medical Reference Project--Status Report.* Western Journal of Medicine. 1986. 145(2). p. 816-822.

21. Wilk S., Michalowski W., and Greco S.. *Supporting Triage of Children with Abdominal Pain in the Emergency Room.* European Journal of Operational Research. 2005. 160(3). p. 696-709.

22. Kawamoto K., Andrew E., and Lobach D.. *Improving Clinical Practice Using Clinical Decision Support Systems: A Systematic Review of Trials to Identify Features Critical to Success.* British Medical Journal. 2005. 330(7494). p. 740-765.

23. Michalowski W. and Kersten M.. *Designing Interactions for Mobile Clinical Decision Support Systems: The MET System Perspective.* MET Research Working Paper MRWP-01-2004. 2004.

24. Michalowski W., Farion K., Pike J., Rubin S., and Slowinski R.. *Development of a Decision Algorithm to Support Emergency Triage of Scrotal Pain and its Implementation in the MET System.* Information Systems and Operational Research. 2005. 43(4). p. 287-301.

25. Nieves M.R.. *Common Causes of Hip Pain in Children.* Pediatrics. 1996. 37(4). p. 787-796. [Available from: http://home.coqui.net/myrna/hip.htm].

26. Shah, S.S.. *Abdominal Gait in a Child With Fever: Diagnosing Septic Arthritis of the Hip.* Pediatric Emergency Care. 2005. 21(2). p. 89-98.

27. MacEwen G.. *The Limping Child.* Pediatrics in Review. 1991. 12(9). p. 268-274.

28. Rao, N.. *Septic Arthritis.* Current Treatment Options in Infectious Diseases. 2002. 4. p. 279-287.

29. Mininder S. Kocher, D.Z., and Kasser J.R.. *Differentiating Between Septic Arthritis and Transient Synovitis of the Hip in Children: An Evidence-based Clinical Prediction Algorithm.* Journal of Bone and Joint Surgery. 1999. 81A(12). p. 86-96.

30. Patil R., Patel-Schneider P., McKay D., FInin T., Gruber T., and Neches R.. *DARPA Knowledge Sharing Effort: Progress Report.* Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning. 1991. Morgan Kaufman. p. 777-788.

31. Gruber, T.. *Toward Principles for The Design of Ontologies Used for Knowledge Sharing.* International Journal of Human-Computer Studies. 1995. 43(5-6). p. 907-928.

32. Mylopoulos, J.. *An Overview Of Knowledge Representation*. On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages. 1983. Springer-Verlag. p. 3-17.

33. Crubezy M., Pincus Z., and Musen M.. *Ontology-Centered Syndromic Surveillance for Bioterrorism*. IEEE Intelligent Systems. 2005. 20(5). p. 26-35.

34. Noy N.F. and McGuinness D.L.. *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford Medical Informatics Report. 2001.

35. Scholten H.. *Ontologies to Structure Models and Modeling Tasks*. 16th JISR-IIASA Workshop on Methodologies and Tools for Complex System Modeling and Integrated Policy Assessment. July, 2002. Laxenburg, Austria.

36. Guarino N.. *Understanding, Building and Using Ontologies*. Human-Computer Studies. 1997. 46. p. 293-310.

37. Devedzic V.. *Understanding Ontological Engineering*. Communications of the ACM. 2002. 45(4). p. 136-144.

38. Perez A.G. *A Survey on Ontology Tool.,* OntoWeb Deliverable 1.3. Ontology-based Information Exchange for Knowledge Management and Electronic Commerce Consortium. 2002. p. 13-34.

39. Yeh I., Noy N.F., and Altman R.B.. *Knowledge Acquisition, Consistency Checking and Concurrency Control for Gene Ontology(GO)*. Bioinformatics. 2003. 19(2). p. 241-248.

40. Heijstf G. and Wielinga B.. *Using Explicit Ontologies in KBS Development.* International Journal of Human-Computer Studies. 1997. 46(2-3). p. 183-292.

41. http://protege.stanford.edu.

42. Schreiber A. Th., Anjewierden A.A., De Hoog R., Shadbolt N.R., Van De Velde W., and Wielinga B.J.. *Knowledge Engineering and Management - The CommonKADS Methodology*. 1999. MIT Press.

43. Gruber T. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal of Human-Computer Studies. 1995. 43. p. 907-928.

44. Horridge M., Rector A., Stevens R., and Wroe C.. *A Practical Guide to Building OWL Ontologies Using The Protege-OWL Plugin and CO-ODE Tools*, Version 1.0. 2004. W3C Recommendation.

45. Crawford J.. *Data Mining in A Scientific Environment.* Australian Unix and Open Systems Users Group 96 & Asia Pacific World Wide Web. September 1996. Victoria, Australia.

46. Steel L.. *Second Generation Expert Systems.* Future Generation Computer Systems. 1985. 1(4). p. 213-221.

47. Geissbuhler A.. *Computer-Assisted Clinical Decision Support.* Decision Making in Health Care. 2003. 14. p. 362-408.

48. Witten I.H., Trigg L., Hall M., Holmes G., and Jo Cunningham S.. *Weka: Practical Machine Learning Tools and Techniques with Java Implementations.* Proceedings of the Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems. 1999. Dunedin, New Zealand. p. 192-196.

49. Protheroe J., Montgomery A., and Peters T.. *Effects of Patients' Preferences on the Treatment of Atrial Fibrillation - Observational Study of Patient-based Decision Analysis.* Western Journal of Medicine. 2001. 174(5): p. 311–315.

50. Lawler F.H.. *Clinical Use of Decision Analysis.* Primary Care. 1995. 22. p. 281-293.

51. Ye N. The Handbook of Data Mining. 2003. Lawrence Erlbaum Associates.

52. Witten I.. *Data Mining -Practical Machine Learning Tools and Techniques with Java Implementations.* 2000. Morgan Kaufmann.

53. Wand, Y. and Weber R.. *An Ontological Model of an Information System.* IEEE Transactions on Software Engineering. 1990. 16(11). p. 1282-1292.

54. Feuer E.J., Kessler E., and Larry G.. *Test Statistic and Sample Size for A Two-Sample McNemar Test.* Biometrics. 1989. 45. p. 629-636.

55. Christopoulou E.. *GAS Ontology: An Ontology for Collaboration among Ubiquitous Computing Devices.* International Journal in Human-Computer Studies. 2005. 62. p. 664-685.

56. Cessie S. and Houwelingen Van J.C.. *Logistic Regression for Correlated Binary Data.* Applied Statistics. 1994. 43(1). p. 95-108.

57. Aleksander I. and Morton H.. An introduction to Neural Computing 2nd edition. 1990. Chapman and Hall.

58. Dietterich T.. *An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization*. Machine Learning. 2000. 40. p. 139-158

59. Oza Nikunj C.. *Ensemble Data Mining Methods*. Encyclopedia of Data Warehousing and Mining. 2005. Idea Group Reference. 1. p 687-698.

60. Jo T. and Japkowicz N.. *Class Imbalances versus Small Disjuncts*. ACM Special Interest Group on Knowledge Discovery and Data Mining Explorations Newsletter. 2004. 6(1). p. 40-49.

61. Grummert E. and Michael K.. *Pediatric Hip Pain: Transient Synovitis versus SepticArthritis*. Journal of Diagnostic Medical Sonography. 2006. 22. p. 185-188.

62. Koop S. and Quanbeck D.. *Three Common Causes of Childhood Hip Pain.* The Pediatric Clinics of North America. 1996. 43(5) p.1053-1064.

63. Provost F. and Fawcett T.. *Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions.* Proceeding of the 3rd International Conference on Knowledge Discovery and Data Mining. 1997. p. 43-48.

64. Kubat M. and Matwin S.. *Addressing the Curse of Imbalanced Data Sets: One-sided Sampling*. Proceeding of the 14th International Conference on Machine Learning. 1997. Morgan Kaufmann. p. 179-186.

65. http://www.pediatric-orthopedics.com/Topics/Perthes/perthes.html.

66. Whitelaw C.C. and Schikler K.N.. *Transient Synovitis.* Emedicine. 2006. [Available from: http://www.emedicine.com/ped/topic1676.htm].

67. Snider R.K.. *Transient Synovitis of the Hip.* Essentials of Musculoskeletal Care, 3rd Edition. 2005. American Academy of Orthopaedic Surgeons. p. 667-668.

68. Guo H. and Victor H.L.. *Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach.* ACM Special Interest Group on Knowledge Discovery and Data Mining Explorations. 2004. 6(1). p. 30-39.

69. Hanley J.A. and McNeil B.J.. *The Meaning and Use of the Area Under the Receiver Operating Characteristic (ROC) Curve.* Radiology. 1982. 143. p.29-36.

70. Ma Y., Guo L., and Cukic B.. *A Statistical Framework for the Prediction of Fault-Proneness*. Advances in Machine Leaning Application in Software Engineering. 2006. Idea Group Publication. p. 128-134.

71. Zoner C.S., Narahashi E., and Honda E.. *Painful Hip in Children*. Brazilian Journal of Rheumatology. 2005. 45(6). p. 66 - 69.

Appendices

Appendix A – Protégé 2000

Protégé 2000 [41] is an ontology development suite to construct domain models and knowledge bases [33]. It was first created by the Medical Informatics Group at Stanford University in 1987. Protégé 2000 is the latest version of Protégé suite. It is a Java-based standalone application and it is oriented to the task of ontology and knowledge-base development. Its original goal was to simplify the knowledge acquisition process for developing expert systems [41], but since then it evolved into a general purpose ontology editing environment. The core of Protégé 2000 is its ontology editor [5, 38, 41], which contains the major modeling functions we have described in section 2.1. It edits the ontology's class taxonomy with class hierarchy and properties attached to classes. It has a variety of plug-ins that support different functionalities, such as reasonning and consistency checking.

Protégé 2000 has the following features:

1) It defines a flexible metaclass architectures and supports many formats such as OKBC, RDF/RDFS and database [41].

2) It can support arbitrary constraints for concepts' properties including cardinality restrictions, default values and inverse properties [55].

3) It supports the CommonKADS methodology.

4) It is designed to be accessible by different applications that rely on ontological models, thus supports interoperability [5].

5) Its user interface is intuitive. It provides a large set of user interface elements for

knowledge modeling.

Figure 8 shows the screen shot of Protégé 2000.

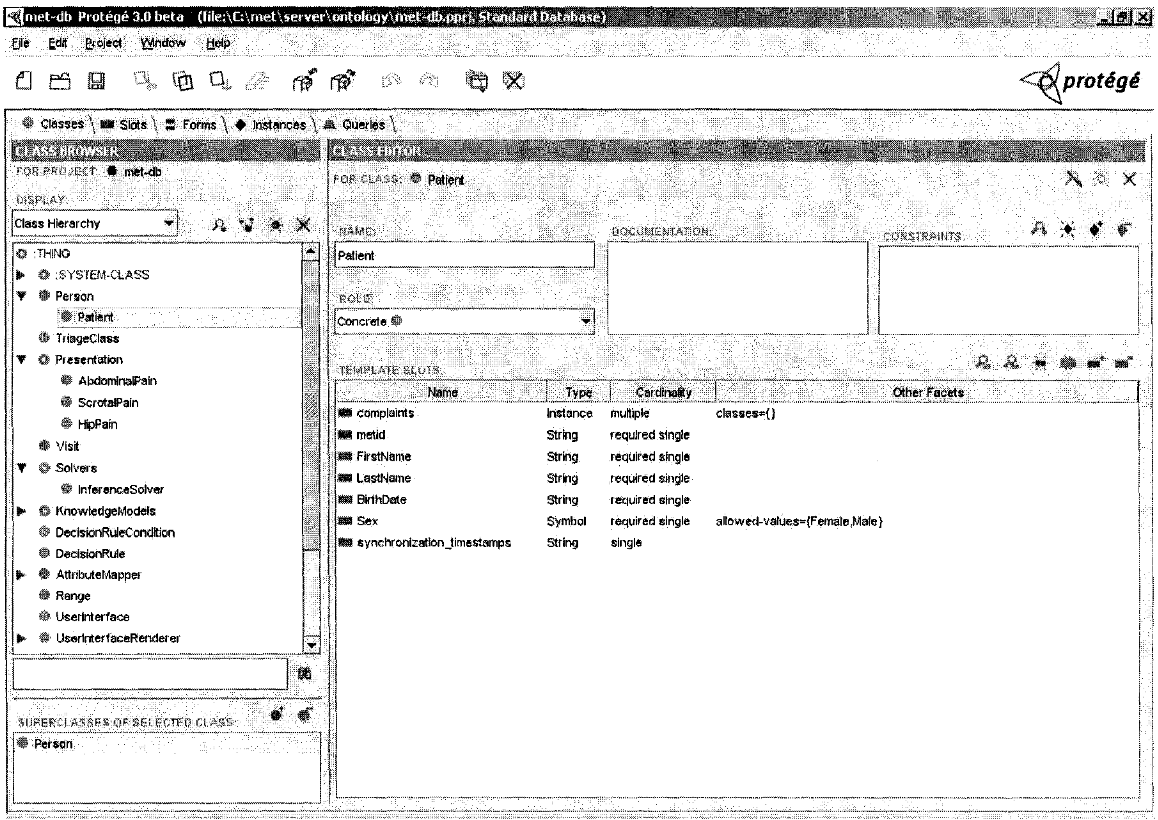


Figure 8 Screen Shot of Protégé 2000

Appendix B – Weka

Waikato Environment for Knowledge Analysis (Weka) [52] is a system developed at the University of Waikato in New Zealand, with a collection of machine learning algorithms for data mining tasks. It contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is written in Java and has been tested under different operating systems. Apart from implementing several data mining algorithms it also has a variety of convenient tools for transforming datasets and preparing training and testing dataset. Detailed information about Weka can be found on its official web site (www.cs.waikato.ac.nz/ml/weka).

One way of using Weka is to preprocess a dataset, apply a learning method to the dataset and analyze its output to extract information about the data [52]. In our research, we used the random dataset split function to prepare our training and testing datasets. We used three learning methods (classifiers) including: J48 tree classifier and Naïve Bayes classifier. Default values of parameters required by each classifier have been specified in Appendix E.

Appendix C – Most Common Incorrectly Classified Patient Records for X-Ray Consult and X-ray_BScan Evaluation

Record Index Attributes	X-Ray Consult				X-Ray_BSCan Evaluation				
	3	41	58	66	10	14	22	25	55
SEX	MALE	FEMALE	MALE	FEMALE	MALE	MALE	FEMALE	MALE	MALE
TEMPERATURE	LT38	?	LT38	?	LT38	GE38	GE38	LT38	GE38
HEART_RATE	GE80_LT120	?	GE120	GE80_LT120	GE40_LT80	GE120	GE80_LT120	GE40_LT80	GE120
AGE	LT6Y	GT6Y	LT6Y	GE6Y	GE6Y	LT6Y	GE6Y	GE6Y	GE6Y
PREV_VISIT	?	YES	YES	?	YES	NO	?	YES	YES
HX_TRAUMA	NO	YES	NO	?	NO	NO	NO	?	?
COMPLAINT_SITE	LEG	HIP	HIP	HIP	LEG	LEG	LEG	HIP	HIP
HX_ILNESS	YES	?	NO	?	NO	NO	YES	YES	?
GAIT_REPORTED	LIMP	LIMP	LIMP	LIMP	NOT_WT_BEAR	NOT_WT_BEAR	LIMP	LIMP	?
DURATION	LT24H	GE7D	GE24H_LT7D	LT24H	GE7D	LT24H	GE24H_LT7D	GE7D	GE24H_LT7D
APPEARANCE	?	WELL_NAD	WELL_NAD	WELL_NAD	WELL_NAD	WELL_NAD	?	WELL_NAD	WELL_NAD
HIP_REST	?	FLEXED_EXT_ROT	?	?	?	?	?	?	?
HIP_ROM	NORMAL	DECREASED	NORMAL	NORMAL	DECREASED	NORMAL	NORMAL	DECREASED	NORMAL
HIP_INT_ROT	NORMAL	DECREASED	NORMAL	NORMAL	DECREASED	NORMAL	NORMAL	?	NORMAL

HIP_FLEXION	NORMAL	DECREASED	NORMAL	NORMAL	DECREASED	NORMAL	NORMAL	DECREASED	NORMAL
GAIT_OBSERVED	NORMAL	?	NORMAL	?	NOT_WT_BEAR	NORMAL	NOT_WT_BEAR	?	LIMP
PAIN_ROM_HIP	NO	YES	NO	?	YES	?	NO	YES	YES
OTHER_PAIN_SITE	NO	?	NO	PELVIS	NO	NO	?	LEG	NO
PAIN_PALPATION	NORMAL	?	NORMAL	TENDER	?	?	TENDER	NORMAL	?
SWELLING	NO	?	NO	?	NO	NO	?	NO	NO
CURRENT_ILNESS	YES	?	YES	NO	NO	NO	NO	YES	NO
PREV_PROBLEMS	NO	NO	NO	NO	NO	NO	NO	NO	NO
WBC	?	?	?	GE12	GE6_LT12	GE12	GE6_LT12	GE6_LT12	GE6_LT12
ESR	?	?	?	LT24	LT24	LT24	GE24_LT40	GE40	LT24

Table 7 Most Common Incorrect Classified Patient Records for X-ray and X-ray/BScan

Appendix D – Missing Values in Testing Dataset

In Table 8 all attributes that have more than 50% of missing values are marked with grey background. They are the following: Previous MD Visit (PREV_VISIT, missing in 65.1% of records), Position of Hip at Rest (HIP_REST, not specified in 80.7% of records!), and laboratory tests including WBC (WBC, missing for 56.6% of records) and ESR (ESR, missing for 56.6% of records).

		% of missing values			
	Attribute	Discharge	X-ray	Lab_Xray_Bscan	Overall
1	SEX	0.0	0.0	0.0	0.0
2	TEMPERATURE	26.2	66.7	0.0	25.3
3	HEART_RATE	7.7	33.3	0.0	8.4
4	AGE	0.0	0.0	0.0	0.0
5	PREV_VISIT	70.8	50.0	41.7	65.1
6	HX_TRAUMA	20.0	16.7	33.3	21.7
7	COMPLAINT_SITE	1.5	0.0	0.0	1.2
8	HX_ILNESS	16.9	50.0	16.7	19.3
9	GAIT_REPORTED	12.3	13.6	25.0	13.3
10	DURATION	4.6	0.0	0.0	3.6
11	APPEARANCE	12.3	33.3	25.0	15.7
12	HIP_REST	78.5	83.3	91.7	80.7
13	HIP_ROM	6.2	0.0	0.0	4.8
14	HIP_INT_ROT	15.4	0.0	25.0	15.7
15	HIP_FLEXION	12.3	0.0	8.3	10.8
16	GAIT_OBSERVED	21.5	50.0	25.0	24.1
17	PAIN_ROM_HIP	16.9	16.7	25.0	18.1
18	OTHER_PAIN_SITE	20.0	16.7	8.3	18.1
19	PAIN_PALPATION	23.1	33.3	50.0	27.7
20	SWELLING	26.2	33.3	41.7	28.9
21	CURRENT_ILNESS	15.4	16.7	0	13.3
22	PREV_PROBLEMS	32.3	0.0	25.0	28.9

23	WBC	66.2	66.7	0	56.6
24	ESR	66.2	66.7	0	56.6

Table 8 Missing Values in Testing Dataset

Appendix E – Data Mining Results

Appendix E-1 DT: J48

Correctly Classified Instances	68	81.9277 %	
Incorrectly Classified Instances	15	18.0723 %	
Total Number of Instances	83		
Three Class Classification Confusion Matrix	64	0	1
	6	0	0
	8	0	4
Two Class Classification Confusion Matrix	4	14	
	1	64	

Table 9 J48 Overall Result

Appendix E-2 DR: LEM2

Correctly Classified Instances	70	84.3373 %	
Incorrectly Classified Instances	13	15.6627 %	
Total Number of Instances	83		
Three Class Classification Confusion Matrix	62	1	2
	4	1	1
	4	1	7
Two Class	10	8	

Classification Confusion Matrix	3	62

Table 10 DR LEM2 Overall Result

Appendix E-3 NB: Naïve Bayes

Correctly Classified Instances	64	77.1084 %	
Incorrectly Classified Instances	19	22.8916 %	
Total Number of Instances	83		
Three Class Classification Confusion Matrix	55	7	3
	3	2	1
	4	1	7
Two Class Classification Confusion Matrix	11	7	
	10	55	

Table 11 NB NaïveBayes Overall Result

Appendix F – Rules That Misclassified Records In X-Ray Consult and X-Ray/BScan Evaluation

Note: A number in brackets following each rule represents its relative strength calculated as the number of training set records that satisfy conditions in a rule and belong to the class indicated by a rule divided by the number of training set records from the indicated class and it is expressed as percentage ratio. Conditions with underline characters in the rule denote missing values in this specific record.

X-ray Consult:

Record 3.

rule 8. (HEART_RATE = GE80_LT120) & (HX_TRAUMA = NO) & (GAIT_REPORTED = LIMP) & (DURATION = LT24H) & (PAIN_ROM_HIP = NO) & (PAIN_PALPATION = NORMAL) => (TRIAGE = DISCHARGE) (7.7)

rule 26. (DURATION = LT24H) & (**APPEARANCE = DISTRESS_UNWELL**) & (HIP_ROM = NORMAL) & (GAIT_OBSERVED = NORMAL) & (**WBC = GE6_LT12**) => (TRIAGE = DISCHARGE) (2.4)

Record 41.

rule 13. (**HEART_RATE = GE120**) & (APPEARANCE = WELL_NAD) & (PAIN_ROM_HIP = YES) & (PREV_PROBLEMS = NO) & (**ESR = LT24**) => (TRIAGE = DISCHARGE) (9.7)

rule 30. (**HEART_RATE = GE40_LT80**) & (**PREV_VISIT = YES**) & (**HX_ILNESS = NO**) & (GAIT_REPORTED = LIMP) & (HIP_INT_ROT = DECREASED) &

(HIP_FLEXION = DECREASED) & (PAIN_ROM_HIP = YES) => (TRIAGE = DISCHARGE) (1.6)

rule 32. (**HEART_RATE = GE120**) & (**HX_ILNESS = NO**) & (PAIN_ROM_HIP = YES) & (PREV_PROBLEMS = NO) & (**WBC = GE6_LT12**) => (TRIAGE = DISCHARGE) (8.1)

rule 65. (SEX = FEMALE) & (**HEART_RATE = GE120**) & (AGE = GE6Y) & (**HX_ILNESS = YES**) & (GAIT_REPORTED = LIMP) => (TRIAGE = LAB_XRAY_BSCAN) (2.9)

rule 73. (**HEART_RATE = GE80_LT120**) & (AGE = GE6Y) & (DURATION = GE7D) & (HIP_INT_ROT = DECREASED) & (**GAIT_OBSERVED = LIMP**) & (PAIN_PALPATION = TENDER) & (PREV_PROBLEMS = NO) & (WBC = GE6_LT12) => (TRIAGE = DISCHARGE) (1.6), (TRIAGE = XRAY) (8.3)

rule 77. (SEX = FEMALE) & (GAIT_REPORTED = LIMP) & (DURATION = GE7D) & (HIP_FLEXION = DECREASED) & (PAIN_PALPATION = NORMAL) & (ESR = LT24) => (TRIAGE = DISCHARGE) (0.4), (TRIAGE = XRAY) (2.1)

Record 58.

rule 16. (HX_ILNESS = NO) & (DURATION = GE24H_LT7D) & (**HIP_REST = NORMAL**) & (GAIT_OBSERVED = NORMAL) & (PAIN_ROM_HIP = NO) & (OTHER_PAIN_SITE = NO) => (TRIAGE = DISCHARGE) (10.1)

rule 18. (SEX = MALE) & (HEART_RATE = GE120) & (GAIT_REPORTED = LIMP) & (PREV_PROBLEMS = NO) => (TRIAGE = DISCHARGE) (8.5)

rule 75. (SEX = MALE) & (AGE = LT6Y) & (HX_TRAUMA = NO) & (HX_ILNESS = NO) & (GAIT_REPORTED = LIMP) & (DURATION = GE24H_LT7D) &

(CURRENT_ILNESS = YES) & (**ESR = LT24**) => (TRIAGE = DISCHARGE) (1.2), (TRIAGE = XRAY) (4.2)

Record 66.

rule 2. (SEX = FEMALE) & (**HX_TRAUMA = NO**) & (HIP_FLEXION = NORMAL) & (PAIN_PALPATION = TENDER) & (**SWELLING = NO**) & (WBC = GE12) & (ESR = LT24) => (TRIAGE = DISCHARGE) (14.2)

rule 9. (SEX = FEMALE) & (AGE = GE6Y) & (HIP_FLEXION = NORMAL) & (**PAIN_ROM_HIP = NO**) => (TRIAGE = DISCHARGE) (4.9)

rule 35. (SEX = FEMALE) & (**PREV_VISIT = NO**) & (GAIT_REPORTED = LIMP) & (PAIN_PALPATION = TENDER) & (CURRENT_ILNESS = NO) => (TRIAGE = DISCHARGE) (10.1)

rule 48. (SEX = FEMALE) & (**HX_TRAUMA = YES**) & (COMPLAINT_SITE = HIP) & (DURATION = LT24H) & (PAIN_PALPATION = TENDER) => (TRIAGE = XRAY) (4.2)

rule 62. (SEX = FEMALE) & (**TEMP = GE38**) & (COMPLAINT_SITE = HIP) & (**HX_ILNESS = YES**) & (HIP_ROM = NORMAL) & (**GAIT_OBSERVED = NOT_WT_BEAR**) => (TRIAGE = LAB_XRAY_BSCAN) (2.9)

rule 76. (**TEMP = LT38**) & (AGE = GE6Y) & (COMPLAINT_SITE = HIP) & (HX_ILNESS = YES) & (HIP_ROM = NORMAL) & (PAIN_PALPATION = TENDER) & (PREV_PROBLEMS = NO) & (WBC = GE12) => (TRIAGE = DISCHARGE) (1.2), (TRIAGE = XRAY) (4.2)

Xray/BScan Evaluation:

Record 10.

rule 22. (AGE = GE6Y) & (COMPLAINT_SITE = LEG) & (HX_ILNESS = NO) & (DURATION = GE7D) & (HIP_INT_ROT = DECREASED) & (PAIN_ROM_HIP = YES) => (TRIAGE = DISCHARGE) (1.2)

Record 14.

rule 7. (AGE = LT6Y) & (HX_TRAUMA = NO) & (COMPLAINT_SITE = LEG) & (APPEARANCE = WELL_NAD) & (HIP_ROM = NORMAL) & (GAIT_OBSERVED = NORMAL) & (**PAIN_ROM_HIP = NO**) & (CURRENT_ILNESS = NO) => (TRIAGE = DISCHARGE) (8.1)

rule 13. (HEART_RATE = GE120) & (APPEARANCE = WELL_NAD) & (**PAIN_ROM_HIP = YES**) & (PREV_PROBLEMS = NO) & (ESR = LT24) => (TRIAGE = DISCHARGE) (9.7)

Record 22.

rule 6. (SEX = FEMALE) & (COMPLAINT_SITE = LEG) & (**APPEARANCE = WELL_NAD**) & (HIP_INT_ROT = NORMAL) & (HIP_FLEXION = NORMAL) => (TRIAGE = DISCHARGE) (15.8)

rule 9. (SEX = FEMALE) & (AGE = GE6Y) & (HIP_FLEXION = NORMAL) & (PAIN_ROM_HIP = NO) => (TRIAGE = DISCHARGE) (4.9)

rule 35. (SEX = FEMALE) & (**PREV_VISIT = NO**) & (GAIT_REPORTED = LIMP) & (PAIN_PALPATION = TENDER) & (CURRENT_ILNESS = NO) => (TRIAGE = DISCHARGE) (10.1)

Record 25.

rule 15. (HEART_RATE = GE40_LT80) & (AGE = GE6Y) & (COMPLAINT_SITE = HIP) & (DURATION = GE7D) & (**HIP_INT_ROT = NORMAL**) => (TRIAGE = DISCHARGE) (1.2)

rule 44. (SEX = MALE) & (HEART_RATE = GE40_LT80) & (COMPLAINT_SITE = HIP) & (**HIP_INT_ROT = DECREASED**) & (OTHER_PAIN_SITE = LEG) => (TRIAGE = XRAY) (4.2)

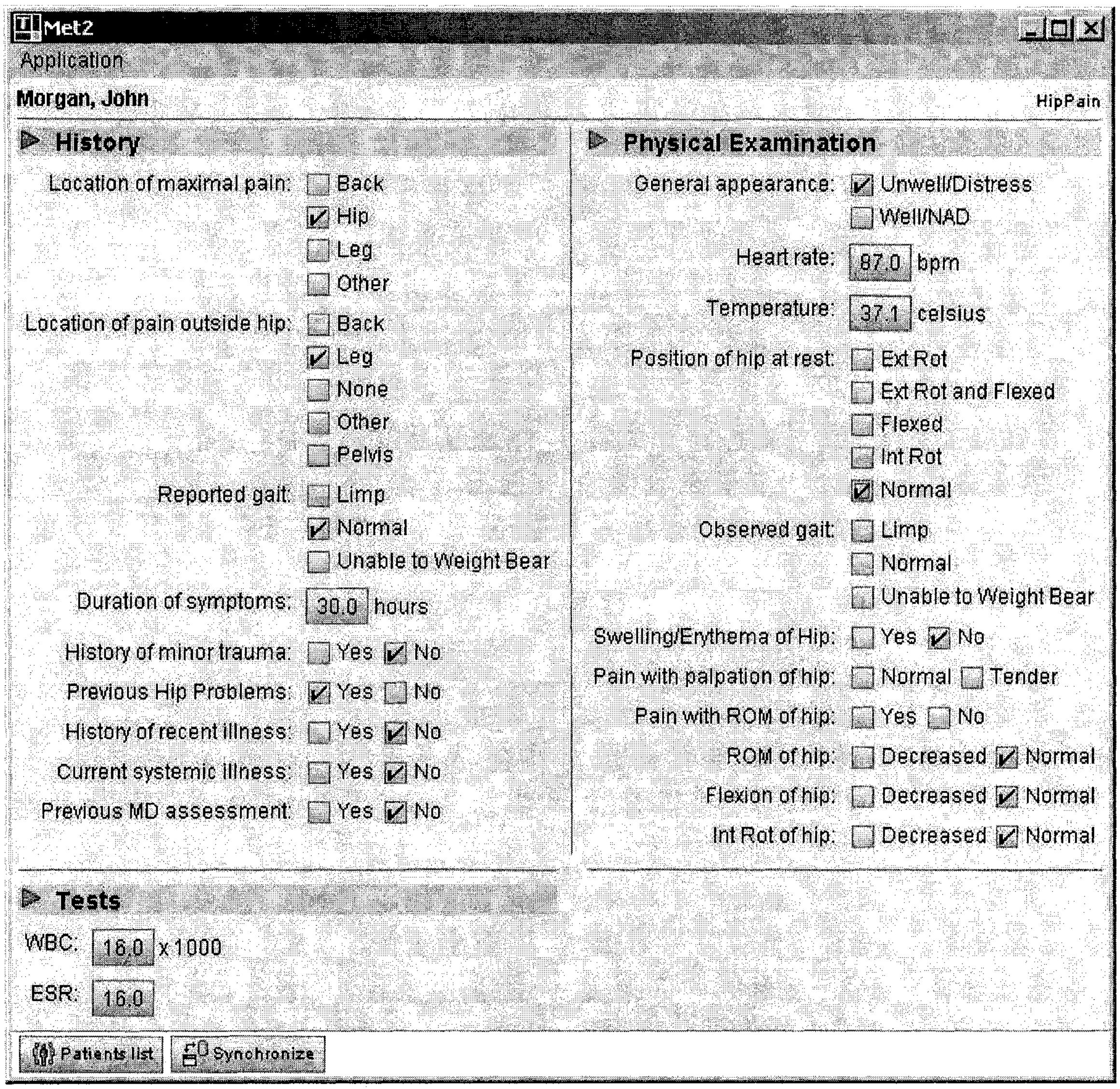
Record 55.

rule 13. (HEART_RATE = GE120) & (APPEARANCE = WELL_NAD) & (PAIN_ROM_HIP = YES) & (PREV_PROBLEMS = NO) & (ESR = LT24) => (TRIAGE = DISCHARGE) (9.7)

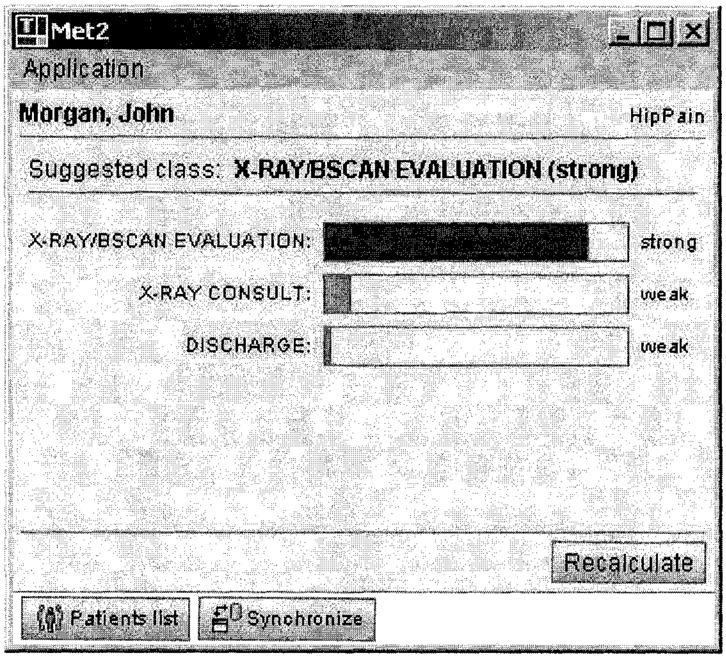
rule 18. (SEX = MALE) & (HEART_RATE = GE120) & (**GAIT_REPORTED = LIMP**) & (PREV_PROBLEMS = NO) => (TRIAGE = DISCHARGE) (8.5)

rule 32. (HEART_RATE = GE120) & (**HX_ILNESS = NO**) & (PAIN_ROM_HIP = YES) & (PREV_PROBLEMS = NO) & (WBC = GE6_LT12) => (TRIAGE = DISCHARGE) (8.1)

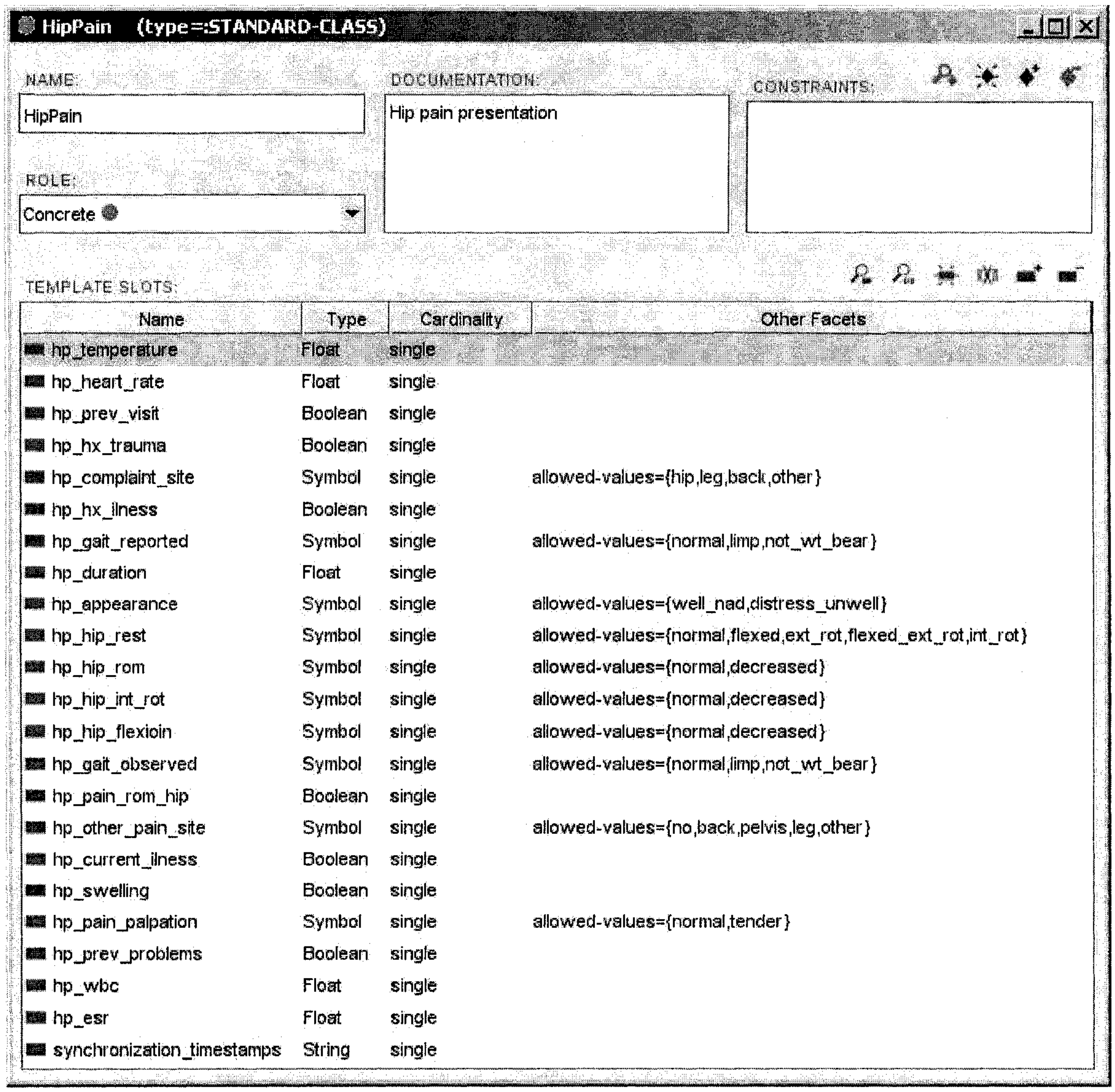
Appendix G – User Interface of the MET-HP for a Desktop Windows-based Computer



Appendix H – User Interface of the MET-HP for Reporting HP Triage Suggestions



Appendix I – Detail of the 'HipPain' Class Implemented in Protégé 2000



Appendix J – Sample Decision Rules from the HP Decision Model

Rule 1:

IF ((HX_ILNESS = NO) & (GAIT_REPORTED = NOT_WT_BEAR) & (DURATION = GE24H_LT7D) & (HIP_ROM = DECREASED) & (HIP_FLEXION = NORMAL) & (OTHER_PAIN_SITE = NO) & (WBC = GE6_LT12))

THEN (TRIAGE = DISCHARGE)

Rule 2:

IF ((SEX = FEMALE) & (HX_TRAUMA = NO) & (HIP_FLEXION = NORMAL) & (PAIN_PALPATION = TENDER) & (SWELLING = NO) & (WBC = GE12) & (ESR = LT24))

THEN (TRIAGE = DISCHARGE)

Rule 3:

IF ((HEART_RATE = GE120) & (AGE = LT6Y) & (COMPLAINT_SITE = HIP) & (GAIT_REPORTED = LIMP) & (DURATION = GE7D) & (HIP_REST = EXT_ROT))

THEN (TRIAGE = XRAY)

Rule 4:

IF ((SEX = FEMALE) & (TEMP = GE38) & (HEART_RATE = GE80_LT120) & (HX_ILNESS = YES) & (HIP_ROM = NORMAL) & (GAIT_OBSERVED = LIMP) & (CURRENT_ILNESS = YES))

THEN (TRIAGE = LAB_XRAY_BSCAN)

Appendix K – Abbreviations Used

Note: Entries in the Table are sorted by the order of the appearance in a body of the text.

Acronym	Value
CDSS	Clinical Decision Support System
HP	Hip Pain
AP	Abdominal Pain
MET	Mobile Emergency Triage
MET-HP	Application implemented in MET for HP presentation
MET-AP	Application implemented in MET for AP presentation
KBS	Knowledge Based System
ODD	Ontology Driven Design
ED	Emergency Department
EPRS	Electronic Patient Record System
ATS	Acute Transient Synovitis
LCPD	Legg-Calve-Perthes Disorder
SA	Septic Arthritis
SCFE	Slipped Capital Femoral Epiphysis
EP	ED physician
CHEO	Children's Hospital of Eastern Ontario
DT	Decision Tree
DR	Decision Rule
NB	Naïve Bayes
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative
ROC	Receiver Operating Characteristic
AUC	Area Under Curve