Published in final edited form as:

Comput Inform Nurs. 2006 ; 24(4): 220–225.

Improving the Human Readability of Arden Syntax Medical Logic Modules Using a Concept-oriented Terminology and Object-oriented Programming Expressions

Jeeyae Choi, R.N., M.S.^{1,3}, Suzanne Bakken, R.N., D.N.Sc.^{1,3,*}, Yves A. Lussier, M.D.^{1,2,*}, and Eneida A. Mendonça, M.D., Ph.D.^{1,*}

¹ Department of Biomedical Informatics, Columbia University, New York, New York

² Department of Medicine, Columbia University, New York, New York

³ School of Nursing, Columbia University, New York, New York

Abstract

Objectives—To develop and evaluate a method for improving the readability of Medical Logic Modules (MLMs) and to discuss the relevance of the method for the creation of MLMs for nursing practice.

Methods—We applied two approaches to improve the human readability of MLMs: 1) exploiting the concept-oriented features of the Medical Entities Dictionary (MED); and 2) building an executable Java program that would be functionally equivalent to the Arden Syntax. Subsequently, we conducted a usability evaluation in which nine participants mapped rules from the Arden Syntax-based MLM to appropriate Java methods in order to reproduce the “screening new or worsening anemia based on intravascular hemoglobin” MLM in Java.

Results—Six of nine (66%) participants mapped 100% of the Arden Syntax rules to Java methods correctly, one mapped 83% correctly, and two mapped 67% correctly. Mistakes included lack of recognition of a rule in the MLM (i.e., rule was not mapped to any Java method) and mapping a rule to an improper Java method. Subjects reported a mapping time ranging from 5–10 minutes. All nine participants described the proposed method as easy to use and easy to understand.

Discussion/Conclusions—The approaches that we have described, along with recent advances in developing concept-oriented representations of clinical data, can play an essential role in the creation of MLMs that are human readable and can potentially increase the number of clinical experts who are willing and able to participate in the creation of MLMs.

Keywords

Guidelines; knowledge representation; decision-making systems; concept-oriented nursing terminology

Introduction

Given the difficulty of knowledge acquisition and representation, knowledge sharing is a significant prerequisite for the efficient development, implementation, and maintenance of knowledge-based systems in health care. Medical logic modules (MLMs) are a procedural

*Faculty authorship is alphabetical. All contributed equally to this research.

representation for sharing task-specific knowledge (i.e., knowledge to support a single decision). A MLM includes three categories of slots: 1) maintenance slots that define name, author, source, version, and validation information; 2) library slots that define links to other sources of knowledge, comments, and keywords for searching; and 3) knowledge slots that define the knowledge [1]. The four main knowledge slots are:

- evoke slot – defines context for MLM (e.g., when falls risk score is stored in the database)
- logic slot – manipulates and analyzes patient data; determines whether further action is required (e.g., determines if patient is at high risk for fall and if action is necessary)
- action slot – defines appropriate action (e.g., send an email alert, store message)
- data slot – maps local terms to terms in MLM

Arden Syntax, which is used by MLMs, is a language for encoding task-specific, procedural knowledge in a manner that is sharable across institutions [1–3]. Although Arden Syntax MLMs are appropriate for representing knowledge such as alerts and reminders of relevance to nursing, there are few descriptions in the published literature [4,5].

Arden Syntax is a Health Level Seven standard and some institutions have widely instituted Arden Syntax-based MLMs, but there are a number of issues that limit their use. First, an Arden Syntax compiler must be written to integrate an Arden Syntax-based decision support system with the institution-specific clinical information system. Second, database schemata, clinical terminologies, and data access methods vary. Therefore, MLMs must be adapted to use the local clinical data repository (CDR). Known as the “curly braces problem” in the Arden community, the use of non-standard expressions, i.e., local terms, is a major barrier to the sharing of MLMs among different institutions [6]. There have been significant efforts toward solving the “curly braces” problem to improve the portability of MLMs. An issue that has received considerably less attention is the human readability of the MLMs. However, this is vitally important because of the essential role of clinical experts in the development and maintenance of MLMs.

In this paper, we describe the development and evaluation of a method for improving the human readability of MLMs. Our method is based upon a concept-oriented terminology and an object-oriented programming language. We apply the method to clinical event monitor MLMs at New York Presbyterian Hospital (NYPH). In addition, although our approaches are broadly applicable, we specifically discuss the relevance of the research to concept-oriented nursing terminologies and the implications of our method for automated processing of task-specific nursing knowledge.

Background

The clinical event monitor at the NYPH monitors data being stored in clinical data repository in order to provide decision support. The system, “Vigilens” is a broad-purpose and server-based decision support system that uses internet and clinical standards to communicate with a) the knowledge-bases in which the alerting conditions are expressed, including controlled terminology servers, b) the relevant clinical data residing on other platforms such as XML and/or HL-7 streams, repositories or warehouses, c) the information technology that provides feedback to users, including web-portals, pagers, emails, clinical systems using asynchronous communication protocols as XML/HL-7 and the IBM Websphere Everywhere middleware offering connection to hundreds of communication protocols and wireless devices (see Figure 1). Vigilens can also take advantage of the tree or directed acyclic graph (DAG) structures of advanced controlled vocabularies in its representation of knowledge; thus alerting rules are simply conceived at the conceptual level and the underlying architecture translates a class-

based alert in a sophisticated query specific to the data structure and database schema or a specific clinical repository. The Medical Entities Dictionary (MED) is a semantic network organized as a DAG that provides an unambiguous representation of clinical data in our central database. Using a DAG such as the MED, Vigilens can query data from multiple sources using nomenclature or classification standards and still uniquely identify the data and the patient to whom they pertain.

There are two types of data represented in curly braces in clinical event monitor MLMs at NYPH: MED codes and Data Access Module (DAM) query language expressions. The clinical event monitor obtains the required MED codes through DAMs. The Arden Syntax is executed based on clinical data. Subsequently, appropriate alerting and warning messages are then sent to clinicians. These processes are performed based on the DAM query expressions within the curly braces. An issue raised by this approach is that, as shown in Figure 2, multiple MED codes and cryptic DAM expressions decrease the human readability of MLMs.

Several researchers have applied object-oriented approaches to solve the curly braces problem in order to increase the shareability and adaptability of MLMs. Karadimas et al. [6] designed an Arden Syntax MLM-based decision support system that uses an object-oriented model and the dynamic linking features of the Java platform. This system demonstrated improved portability of MLMs across institutions. Peleg and colleagues [7] developed an expression language, called GEL, based on Arden Syntax's logic grammar. GEL is designed to reconcile the differences between Arden Syntax's data model and an object-oriented data model. Although it fills the gap between procedural and object-oriented data models, it does not directly support the object-oriented model. This limitation is addressed by GELLO, an object-oriented expression language that supports harmonization with object-oriented models [8]. Although these studies do not directly address the human readability issue, we hypothesize that clinicians may perceive such object-oriented expressions as easier to read than procedural rules such as those in Arden Syntax-based MLMs.

Methodology

We applied two approaches to improve the human readability of the MLMs: 1) exploiting the concept-oriented features of the MED; and 2) building an executable Java program that would be functionally equivalent to the Arden Syntax. Subsequently, we conducted a usability evaluation of these approaches.

Exploiting Concept-oriented Features of the MED

Clinical data that are used in clinical event monitors are defined in the MED that contains more than 69,000 medical concepts that are used at NYPH [9]. The concepts in the MED are represented as frames and are arranged in a semantic network. The semantic network supports IS-A and associative relationships among concepts. Each frame includes information specific to a particular concept, such as its name, synonyms, units of measure when applicable, etc. To reduce the number of MED codes displayed within the curly braces, we applied Tao et al.'s class-based representation of the MED and an associated visualization tool, the Ontology Browser System (OBS) [10]. The OBS graphically displays the hierarchical relations of multiple terms, which facilitates the identification of the common or nearest common ancestors for selected MED codes (See Figure 2) [11]. The potential impact of this for human readability is that it is possible to include only the MED code for the nearest common ancestor rather than all MED codes for the clinical data required for the clinical event monitor MLMs.

Java Expression Approach

We examined three different techniques that might be used to translate the current Arden Syntax-based MLMs into object-oriented language: 1) building an executable Java program that is functionally equivalent to the Arden Syntax; 2) constructing a robust object-oriented design in Java that is not completely functionally equivalent to the Arden Syntax; and 3) developing both a functional equivalent and a reliable, maintainable, shareable, and scalable Java program. The second approach was rejected because preservation of the functionality of the original program was deemed to be very important. The third approach was considered to be the best in the long-term, but given that our primary purpose in this evaluation was to improve the human readability, the first was chosen for its simplicity and cost-effectiveness.

An object-oriented software architecture for representing MLM rules was independently designed and developed using unified modeling language (UML). The programmer wrote executable Java programs that could translate simple functions from Arden Syntax-based MLMs to the object-oriented Java programming language. To clarify the specific meaning of the DAM query expressions so that items could be more accurately mapped to Java, we transformed the query expressions into pseudo code (Figure 3).

Usability Evaluation

For the usability evaluation, we first rewrote a MLM, “screening new or worsening anemia based on intravascular hemoglobin,” based on Arden Syntax 2.0. Then we selected a set of six functionally-equivalent Arden Syntax rules and Java methods and organized them into a table. The sample included eight clinicians (five nurses and three physicians - one with computer science training) and one computer scientist; eight of the nine participants were familiar with Arden Syntax MLMs. We asked participants to map rules from the Arden Syntax-based MLM to appropriate Java methods in order to reproduce the “screening new or worsening anemia based on intravascular hemoglobin” MLM in Java.

Results

Six of nine (66%) participants mapped 100% of the Arden Syntax rules to Java methods correctly, one mapped 83% correctly, and two mapped 67% correctly. Mistakes included failing to recognize a rule in the anemia_monitor MLM (i.e., rule was not mapped to any Java method) and mapping a rule to an improper Java method. Subjects reported a mapping time ranging from 5–10 minutes. All nine participants described the proposed method as easy to use and easy to understand. Positive comments included: “Quite straightforward,” “It didn’t take me a long time to map things up,” “There was no difficulty in mapping,” and “The MLM became so easy to read after mapping.” Negative comments included: “Some Java rules were a bit hard to understand (e.g., ProcCompDAO)”.

Discussion

Study results provide evidence that our method improved the readability of the study MLM. Moreover, the results suggest that participants were able to apply the Java-based expressions with a high degree of accuracy and efficiency. Although not the focus of our study, our method has the potential to decrease the burden of designing and implementing a compiler due to the simplification of codes and improved clarity of query expressions.

There were a number of limitations to our study. First, we used a single MLM. Second, the majority of the participants were familiar with Arden Syntax MLMs. Third, there were six Arden Syntax rules and six JAVA expressions, and consequently some matches may have

occurred by process of elimination. Despite these limitations, the findings are promising and applicable beyond our institution.

Although not widely implemented for nursing practice, MLMs are appropriate for representing task-specific nursing knowledge. Examples include:

- Reminder to change IV tubing per institutional protocol
- Alert that patient is at high risk for pressure ulcer
- Notification that patient qualifies for urinary incontinence clinical pathway
- Suggestions for assessing willingness to quit in person identified as a smoker and short-term smoking cessation interventions
- Alert regarding discharge teaching for patient and family members

The methods that we developed and tested will facilitate the development of MLMs for nursing tasks such as these in two ways.

First, the class-based approach to reducing the number of MED codes required within the curly braces of the MLM can be generalized to other concept-oriented terminologies. During the last decade there has been tremendous effort toward the creation of concept-oriented representations of nursing terms. This includes the representation of the International Classification of Nursing Practice using the Galen Representation and Integration Language (GRAIL) [12], the development of International Standards Organization (ISO) reference terminology models for nursing diagnoses and nursing interventions [13], and the integration of nursing diagnoses, interventions, and outcomes into SNOMED Clinical Terms [14]. These concept-oriented representations enable a class-based approach to reducing the number of codes required for the clinical data in MLMs. This is in addition to the benefit gained from use of standardized nursing terminologies to reduce the “curly braces” problem and to consequently increase the portability of MLMs.

Second, clinical nursing expertise is essential to the creation of nursing MLMs. Consequently, approaches such as the Java-based method that we developed, which participants perceived to be understandable and useful, i.e., more human readable, may be helpful in engaging nurses to participate in the development of MLMs.

Conclusions

There has been little implementation of MLMs to encode task-specific knowledge for decision support in nursing practice. The approaches that we have described, along with recent advances in developing concept-oriented representations of clinical data, can play an essential role in the creation of MLMs that are human readable and can potentially increase the number of clinical experts who are willing and able to participate in the creation of MLMs. This is particularly vital for nursing given the small number of nurses with formal education in informatics.

Acknowledgments

The authors acknowledge the support and advice of Professor Jhon Brackett, Ying Tao, and Dr. Dongwen Wang. The research was supported by 528753/PO P417322 grant from NASA/UVC. Preparation of the manuscript was supported by 5P20NR007799 and 1R01NR008903 from the National Institutes of Health.

References

1. Hripcsak G, Ludemann P, Pryor TA, Wigertz OV, Clayton PD. Rationale for the Arden Syntax. *Computers and Biomedical Research* 1994;27(4):291–324. [PubMed: 7956129]
2. Wigertz, O.; Hripcsak, G.; Shahsavar, N.; Bagenholm, P.; Ahlfeldt, H.; Gill, H. *Knowledge and Decisions in Health Telematics*. IOS Press; 1994. *Data-Driven Medical Knowledge-Based Systems Based on Arden Syntax*; p. 126-131.
3. Bang, M.; Eriksson, H. *Generation of Development Environments for the Arden Syntax*. Department of Computer Science, Linköping University; 1997.
4. Liang, YC.; Chang, P. The development of variable MLM editor and TSQL translator based on Arden Syntax in Taiwan. *AMIA Annu Symp Proc*; 2003; 2003. p. 908
5. Campbell JR, Stoupa R, Warren JJ. Design and implementation of a rule based system for ambulatory nursing data management. *Proc Annu Symp Comput Appl Med Care* 1991:659–63. [PubMed: 1807685]
6. Karadimas HC, Chaillolleau C, Hemery F, Simmonet J, Lepage E. Arden/J: An Architecture for MLM Execution on the Java Platform. *JAMIA* 2002;9(4):359–368. [PubMed: 12087117]
7. Peleg, M.; Ogunyemi, O.; Tu, S.; Boxwala, AA.; Zeng, Q.; Greenes, RA.; Shortliffe, EH. *AMIA Annual symposium*. Washington, DC: Conference Proceedings; 2001. *Using Features of Arden Syntax with Object-Oriented Medical Data Models for Guideline Modeling*.
8. Denekamp, Y.; Ogunyemi, O.; Boxwala, AA.; Greenes, RA. *AMIA Annual Symposium*. San Antonio; Texas: 2002. *Using a New Object Oriented Expression Language (GELLO) to Encode Arden Syntax Medical Logic Modules*.
9. Cimino J, Clayton P, Hripcsak G, Johnson S. Knowledge-based approaches to the maintenance of a large controlled medical terminology. *JAMIA* 1994;1(1):35–50. [PubMed: 7719786]
10. Tao Y, Mendoca EA, Lussier YA. A Tool to Facilitate the Analysis of Medical Terminologies. *AMIA*. 2003
11. Tao, Y.; Mendoca, EA.; Lussier, YA. submitted to *MedInfo*. Sanfrancisco, CA: 2004. *Common Ancestry Summarizer: A Tool for Abstracting Relevant Classes of Concepts*.
12. Hardiker NR, Rector AL. Modeling nursing terminology using the GRAIL representation language. *Journal of the American Medical Informatics Association* 1998;5(1):120–128. [PubMed: 9452991]
13. Bakken S, Coenen A, Saba V. ISO reference technology model: nursing diagnosis and action models look to testing for practical application. *Healthcare Informatics* 2004;21(9):52. [PubMed: 15457878]
14. Bakken S, Warren JJ, Lundberg C, Casey A, Correia C, Konicek D, Zingo C. An evaluation of the usefulness of two terminology models for integrating nursing diagnosis concepts into SNOMED Clinical Terms. *International Journal of Medical Informatics* 2002;68(1–3):71–7. [PubMed: 12467792]

Figure 1.
Architecture for Clinical Event Monitor (say more)


```

knowledge:
type: data-driven;;
data:
  potassium_storage := event {'32506','1805';
                              '32506','40469';
                              '32506','33801';
                              '32506','2477';
                              '32506','28186';
                              '32506','40435';
                              '32506','28094';
                              '32506','28093';
                              '32506','35589';
                              '32506','35741';
                              '32506','35409'};

  raw_potassiums := read last 3 from (
    {'evoking','dam'="PDQRES2",'constraints'="C*****"; ;
     '32310'});
  potassium := last (raw_potassiums where they are number);
  diuretic_codes := read
    {'dam'="PDQORD1",'auxstr'="0013",'constraints'="C*****",
     'status_value'="A",'display_header'="T",'display_comp'="O"; ;
     '31062','31092','31095','31173','31190','31210','31269'};
  hypokalemia_boundary := 3.2;
  last_alert := read last
    ({'dam'="PDQDEC1",'display_header'="TX",'display_comp'="";
     'mlmself','mlm DIURETICS_AND_HYPOKALEMIA'}
    where it occurred within the past 3 months);

```

Figure 2.

Excerpt from a clinical event monitor MLM related to potassium levels and non-potassium sparing diuretics. Note the multiple MED codes for potassium levels and the cryptic DAM query expressions.

Figure 3.

Sample OBS output showing the common ancestors (93 and 144) and the nearest common ancestor (32948) for MED codes (33892 and 34651).

```

/*****
/* Get the last VALID hemoglobin AND Define valid previous */
/*hemoglobin from the raw data (as some stored hemoglobins may */
/* or may not be valid). */
*****/
previous_hemoglobin := read last 1 from
{ Query the repository using DB2 connect or the replicator to find
  the next most recent hemoglobin (not the one that evoked the
  MLM) that also has a numerical value (it seems that some
  hemoglobins stored in the repository don't have values!) using
  this code and all descendants: '32171'};
```

Figure 4.

A partial view of pseudo code written within the curly braces

```

/*****/
/* Get the data that evoked this MLM */
/*****/
current_hemoglobin := read last {'evoking', 'dam'="PDQRES2",
                               'constraints' = "C*****"; ;'32171'};
/*****/
/* Get the last 11 hemoglobins including the current one */
/* (may or may not be valid). */
/*****/
raw_hemoglobins := read last 11 from
{'dam'="PDQRES2", 'constraints' = "C*****"; ; '32171'};

```

Figure 5.

An example of a DAM query expression in Arden Syntax-based MLM

```
/* **** */
/* Get the last VALID hemoglobin */
/* AND Define valid previous hemoglobin from the raw data . */
/* (as some stored hemoglobins may or may not be valid). */
/* **** */
previous_hemoglobin := read last 1 from {AM.getLastValue()};
```

Figure 6.

A partial view of anemia_monitor MLM with a Java method in the curly braces.