Thesis for the Degree of Doctor of Philosophy

SEMANTIC RECONCILIATION MODEL FOR SHAREABLE AND INTEROPERABLE MEDICAL KNOWLEDGE ACQUISITION

Taqdir Ali

Department of Computer Science and Engineering
Graduate School
Kyung Hee University
South Korea

February 2019

SEMANTIC RECONCILIATION MODEL FOR SHAREABLE AND INTEROPERABLE MEDICAL KNOWLEDGE ACQUISITION

Taqdir Ali

Department of Computer Science and Engineering
Graduate School
Kyung Hee University
South Korea

February 2019

SEMANTIC RECONCILIATION MODEL FOR SHAREABLE AND INTEROPERABLE MEDICAL KNOWLEDGE ACQUISITION

by

Taqdir Ali

Supervised by

Prof. Sungyoung Lee

Submitted to the Department of Computer Science and Engineering and the Faculty of Graduate School of Kyung Hee University in partial fulfillment of the requirements of the degree of Doctor of Philosophy

Dissertation Committee:

Prof. Oksam Chae ______________________

Prof. Tae-Seong Kim ______________________

Prof. Wajahat Ali Khan ______________________

Prof. Maqbool Hussain ______________________

Prof. Sungyoung Lee ______________________

Achievement of each challenging goal is possible due to self-efforts and the guidance of elders, especially those who were close to our heart.

My humble efforts, I dedicated to my beloved

Father (may ALLAH rest his soul in peace) & Mother,

Brothers & Sisters,

My beloved wife & Kids

whose endless love, encouragement, support, and prays make me able to achieve such success and honor.

Along with my all respected, hardworking, and supportive

Teachers.

Abstract

Technologically integrated healthcare systems can be realized if physicians are encouraged to use smart systems in different phases of patient care such as diagnosis, treatment, and follow-ups. Clinical decision support system (CDSS) plays an important role in decision making. The knowledge base is core component of CDSS. However, adaption of CDSS in clinical work-flows is challenging due to the knowledge base evolution according to continuous innovative research in medical domain. The existing clinical data, physician's heuristics, experiences, and practices are considered as contributing resources to the knowledge base evolution. Currently, CDSS is heading towards smart environments but lack the support, for abstraction of technology oriented knowledge, from physicians, and also lack the shareable and interoperable knowledge creation environment. Therefore, abstraction in the form of user-friendly and flexible authoring environment is required for physicians, in order to smartly create shareable and interoperable knowledge for CDSS work flows.

Existing CDSS systems lack interoperability and shareability of knowledge due to avoiding medical standards. The utilization of medical standards increases the knowledge creation complexity and overburdens the physicians to evolve the knowledge. Therefore, we proposed a *semantic reconciliation model (SRM)* to create shareable and interoperable knowledge using a user-friendly authoring environment. Firstly, the proposed model provides *schema-data level semantic reconciliation* using flexible mapping methodology to achieve the knowledge interoperability goal. Secondly, the SRM provides *structure level semantic reconciliation* to create shareable knowledge. The convergence of medical standards can make the knowledge shareable and interoperable, and fortunately, medical domain is rich for standards.

The SRM model achieves knowledge shareability using standard representation such as Med-

ical Logic Module (MLM) using HL7 Arden Syntax. Each standard representation of knowledge has specific structure and syntax, therefore, *structure level semantic reconciliation* deals with structure and syntax of knowledge representation. On the other hand, the interoperability aspect is handled with *schema-data level semantic reconciliation*, which provides mappings among standard and non-standard terminologies and standard data models. The amalgamation of standard terminologies and data models into standard knowledge representation resolves the aforementioned CDSS limitations with respect to knowledge shareability and interoperability. However, creating shareable and interoperable knowledge using Arden syntax without abstraction increases complexity, which ultimately make the process difficult for physicians to use authoring environments. Therefore, Physician friendliness is provided by abstraction at the application layer to reduce the complexity. We designed and developed an *Intelligent-Knowledge Authoring Tool* (I-KAT) to realize the SRM methodology for providing abstraction and hiding the structural and syntactic complexity from physicians. This abstraction is regulated by the mappings created between legacy system concepts, which are modeled as domain clinical model (DCM) and medical standards such as virtual medical record (vMR) and Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT).

The multi-modal mappings *schema-data level semantic reconciliation* is a prerequisite for interoperable knowledge acquisition method. The *schema-data level semantic reconciliation* provides mapping algorithms among non-standard terminology such as DCM, standard terminology such as SNOMED CT, and standard data model Virtual Medical Record (vMR). Existing mapping algorithms achieved remarkable accuracy in standard ontologies mapping but they only focused on internal semantics within ontologies. We inset explicit semantics before calculating similarity score, which enhances the mappings accuracy. Similarly, our proposed mapping algorithm also deals with definition based mapping of concepts that shows the productive results with respect to accuracy.

In multi-model mapping algorithms, our proposed algorithm for *DCM-Standard Terminology Mapping* is evaluated with state-of-the-art systems with statistical measures such as precision, recall, and F-Measure. Our algorithm shows better results than the existing systems as precision 0.95, recall 0.92, and F-Measure 0.93. Usually, the ontology matching algorithms lack definition based algorithm, therefore, we evaluated our proposed definition based algorithm with base-line (Jaccard) algorithm for *Standard Terminology and Data Model Mapping*. As compared to base-line (Jaccard) algorithm, our proposed algorithm showed better results as precision (0.89), recall (0.97), and F-measure (0.93).

As the objective of the study is creation of shareable and interoperable knowledge using a user-friendly and flexible I-KAT. Therefore we evaluated our system using the completeness and user satisfaction criteria. We assessed the system through the system and user centric evaluation processes. For system centric evaluation, we compared implementation of clinical information modelling systems requirement in our proposed system and in existing system. The results suggested that 82.05% of the requirements were fully supported, 7.69% were partially supported, and 10.25% were not supported at all by our system. Whereas, in the existing system, 35.89% are fully supported, 28.20% are partially supported, and 35.89% are not supported at all. In user centric evaluation the assessment criterion was 'ease of use'. The proposed system showed 15 times better results with respect to time in MLM creation as compare to the existing systems. Moreover, the participants on average made only one error in MLM creation using our proposed system, while the average rate of error using existing systems was 13 per MLM. We also evaluated the proposed system efficiency with respect to time on task and task success rate. The task success rate of the proposed system was 90.62%, while the existing system was 46.87%. Based on the ratio of mean time completion to the success rate, the overall efficiency of the proposed system was 56.62%, which is better than the existing system efficiency 1.85% completion rate/time.

We provide a user-friendly authoring environment for creation of shareable and interoperable knowledge for CDSS to overcome the complexity of knowledge acquisition. The authoring environment uses state of the art decision support related clinical standards with increased ease of use.

Acknowledgement

First and foremost, I render my humble and sincere thanks to the ***Almighty Allah*** for showering HIS blessings upon me. The Almighty gave me the strength, courage, and patience during my doctoral study.

Special thanks to my advisor Prof. Sungyoung Lee who provided me guidance, strength, support, and courage in overcoming the difficult challenges throughout my time as his student. I have learned a lot from him in becoming a productive person in diverse situations. He inspired me with his dynamic personality and his unreserved help and guidance lead me to finish my thesis. He has great role in polishing my skills such as thinking, creativity and technical soundness which are key ingredients for high quality research.

I am grateful to my dissertation evaluation committee for their insight comments and valuable suggestions during the dissertation defense. Their valuable comments improved the presentation and contents of this dissertation.

I am extremely grateful to some of my colleagues who have always provided me time, expertise, and encouragement in my course of research. They were always present to guide me in difficult situations of my PhD duration. I would like to thank Dr Asad Masood Khattak, Dr Zeeshan Pervaiz, Dr. Wajahat Ali Khan, Dr. Bilal Amin, Dr. Maqbool Hussain, Dr. Muhammad Afzal, Dr. Rahman Ali, Shujaat Hussain, and Jamil Hussain. They have contributed enormously in successfully performing various academic and personal tasks that confronted me during my stay at South Korea.

I am very thankful to all of my current and former Ubiquitous computing lab fellows and colleagues for their kind support to my personal and academic life at Kyung Hee University. I am highly obliged to brilliant researchers Tae Ho Hur, Jae Hun Bang, Bilal Ali, Asif Razzaq, Usman

Akhtar, Imran Ali, Musarrat Hussain, Muhammad Sadiq, Mrs. Seoungae Kim, Dr. Kifayat Ullah, Omar Farooq, Dr. Mahmood Ahmad, Saeed Ullah, Dr. Ahsan Raza Kazmi, and Zain Abbas. This journey would not have been possible without their support. They contributed a lot in to my personal and academic life to polish myself. Also, I appreciate all my Korean and international friends who worked as a team with me and developed my team work skills and provided me wonderful memories during my stay in South Korea.

Last but not the least, I would like to express my sincere gratitude to my mother, sisters, and brothers for their endless love, support, prayers, and encouragement. Their support and encouragement has made this dissertation possible. I would like to thank my wife Mrs. Farzana Raza and Kids (Ibaad Ulhassan, Dua Zahra, and Rumeha Ali) for their love, support and guidance in every difficult situation I faced during all this time. Finally, I would like to leave the remaining space in memory of my father for his encouragement in all of my pursuits and inspiration to follow my dreams.

Taqdir Ali

October, 2018

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

A clinical decision support system (CDSS) provides an effective and proficient service model in the healthcare domain to enhance service quality and cost effectiveness of patient care [3]. CDSS have been used to improve the patient care quality, reduce the chance of decision making errors, save physicians' time, and reduce medical costs [4]. A recent study has shown the enhancement of physicians' performance using CDSS in diagnosis 4.8 - 33.9% for obesity, 2.3 - 11.9% for smoking, 1.1 - 4.6% for pediatric depression, and 0.2 - 8.8% for adult depression [5]. According to a similar CDSS survey [6], CDSS enhanced the physicians' performance up-to 40% by diagnostic systems, 76% by reminder systems, 62% by disease management systems, and 66% by treatment and prescription systems. The Department of Health and Human Services (HHS) of the Office of National Coordination (ONC) considered the clinical decision support as an important feature for the certified electronic health records (EHR) [7]. Physician adaptation to CDSS depends on the user-friendliness aspect of the system. CDSS support and assist physicians in making right decisions at right time during patient care [3, 8]. CDSS creates a coordination path between patients and physicians by providing effective recommendations, alerts, and reminders at the proper time [9]. A number of methods for automatic diagnosis, treatment, and medication administration are proposed in order to support the process of clinical decision-making [10–12]. For effective recommendations and alerts, CDSS requires a knowledge base to regulate the system's information flow.

CDSS is comprised of three main components: knowledge base, inference engine, and communication mechanism [13]. The key factors for success of a CDSS are knowledge quality and continuous improvement of that knowledge [14]. In literature, various barriers about knowledge, have also been considered to the CDSS adoption such as resource deficiency to create and maintain knowledge, lack of standard knowledge representation, integration difficulties of CDSS in clinical workflows, and dissemination of knowledge [15]. Addition to these barriers, another aspect is the ignorance of knowledge creation stakeholders - Domain Experts (Physicians) to facilitate with user-friendly authoring environment for knowledge acquisition. The system interfaces complexity of knowledge acquisition overburden and move away the physicians from evolving knowledge base with their heuristics, practices and experiences in the real field. According to a group study with physicians' [16], they feel knowledge authoring as a tedious task using complex system interfaces, which resists their reasoning capability and loses their autonomy. The most important and frequently reported issues with CDSSs are integration into clinical workflows and dissemination of successful interventions from one system to another [17]. However, the fundamental barriers to successful utilization of CDSSs are creating, enhancing, and managing the knowledge base [18] and disseminating the created knowledge [19]. One solution for removing these barriers is to create a shareable and interoperable knowledge base. While this eliminates barriers on one hand, it incurs knowledge acquisition complexity on the other hand, which again drives physicians away from using CDSSs.

In aforementioned barriers, we focus on three aspects shareability of knowledge, interoperability of knowledge, and user-friendliness of the system to create shareable and interoperable knowledge.

1.1.1 Knowledge Shareability

The medical domain has many clinical standards for data, knowledge, and communication. Therefore, knowledge shareability can be realized through a standard representation of the knowledge. The Health Level-7 (HL7) community has designed and developed Medical Logic Module (MLM) using a specific language Arden Syntax as a standard unit of medical knowledge for healthcare [20]. HL7 Arden Syntax is an ANSI standard and provides a comprehensive structure for representing clinical knowledge [21, 22]. The main intention of Arden Syntax is to enable physicians to easily transform their clinical real practices and experiences into sharable knowledge [22]. Arden syntax supports a large number of various operators, control structures, decision, and looping structures and comprehensive data types [2]. The knowledge engineers are responsible to transform the physicians' heuristics and knowledge into corresponding Arden Syntax MLM using required Arden Syntax artifacts and provides training to the physicians to be able for MLM composing without knowledge engineers intervention. However, it overburdens the physicians to compose MLM by themselves and it leads to a challenge of transforming the clinical knowledge into MLMs [23]. The existing systems [24] have complex interfaces for composing MLMs along with its complex structure and syntax, which create a significant barrier for knowledge acquisition.

1.1.2 Knowledge Interoperability

To realize the effectiveness of knowledge shareability, the knowledge base should be capable to integrate with any clinical workflows in easy manner. The knowledge interoperability can achieve with utilization of medical standards. Currently, Knowledge acquisition tools have not been successfully adopted the knowledge interoperability due to the minimal support of standard data models [14]. Consequently, the existing knowledge acquisition tools fail to resolve the heterogeneity problem of clinical information models [25]. Another intrinsic barrier for creating MLMs is the curly brace problem of querying the required input data from the medical systems. The CDSS community has recommended a standard information model, virtual medical record (vMR), to resolve the issue of heterogeneous data models for CDSS systems [26] as well as the curly brace problem [27]. The standard data model vMR meets the scalability and interoperability objectives for a knowledge acquisition tool [26, 28]. Knowledge that is created via standard CDSS input and output specified in the vMR data model is easily integrated among different CDSS systems. However, the use of the standard data model vMR needs to link with a standard terminology to maximize system interoperability. For example, Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT) [29] supports comprehensive terminology in the clinical domain used by physicians worldwide [30, 31]. Incorporating the standard data model vMR into Arden Syntax MLM increases the complexity of clinical rules for domain experts who will need to learn the technical specifications of the standard. Additionally, physicians are comfortable with local terminologies and prefer creating rules using localized concepts instead of standard terminologies due to maximum recall of concepts.

1.1.3 User Friendliness

According to the aforementioned features of knowledge, the convergence of standard knowledge representation, standard data models, and standard terminologies into a single knowledge acquisition platform enhances the knowledge creation, modification, and validation for end users. On the other hand, mostly physicians complaint about complex interfaces of healthcare systems, which drastically decrease the physicians performance [16] and consider as a prominent barrier in CDSS adoption. The structure and syntax of standard knowledge representation is tedious and overburden the physicians in knowledge creation process. In similar way, the standard data model classes and their attributes with corresponding concepts of standard terminologies are hard to remember and recall during knowledge authoring. Usually physicians also prefer to utilize the localized medical concepts instead of standard terminologies concepts. Therefore, we propose a model to deal with knowledge shareability and interoperability with high level abstraction of technology-oriented knowledge authoring environment.

1.2 Motivation

In the aforementioned sections, we described that existing systems lack shareable and interoperable knowledge base due to minimal utilization of standard knowledge representations, data models, and terminologies. The dissemination of the experts' heuristics, practices, and experiences with community is cost effective, saves the time of physicians, and beneficial for novice practitioners. Along with shareability aspect, the interoperable knowledge can be easily integrated to legacy healthcare systems and clinical workflows. These features can be implemented using standard knowledge representation, data models, and terminologies. The utilization of standards achieve these features on one hand, but increase the knowledge creation complexity. The tradeoff among shareability, interoperability, and user-friendliness is shown in Figure 1.1

Knowledge creation complexity makes the physicians dependent on the knowledge engineers. Therefore, existing systems have strong dependency on the knowledge engineers to create knowledge with the input from physicians. The dependency on knowledge engineers causes incomplete knowledge transfer, understanding dilemma due to communication gap, and additional cost of hir-

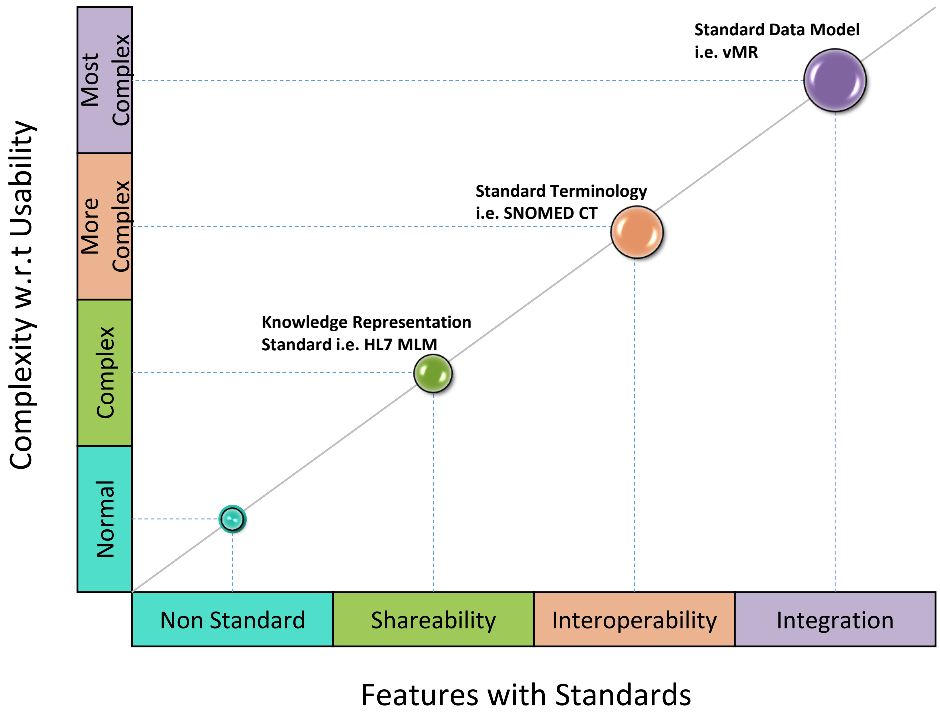

Figure 1.1: Motivational tradeoff among shareability, interoperability and User-friendliness

ing knowledge engineers. In existing systems, physicians need training for standard languages of knowledge representation such as Arden Syntax. It reduces dependency on knowledge engineers, but difficulty level of these specialized languages prompted physicians to limit relying on these systems. However, physicians require a simple, understandable, and manageable knowledge acquisition methodology to create knowledge in easy manner. Therefore, an easy knowledge creation methodology with user-friendly interfaces, with maximum abstraction of technical aspects, and closeness to localized terminologies, is the most suitable system for physicians. The system should have aspects of knowledge shareability and interoperability, which would attract physicians to adopt the CDSS with knowledge acquisition and maintenance process for shareable and interoperable knowledge.

1.3 Problem Statement

The knowledge acquisition is an important aspect and process of clinical knowledge base evolution. Diverse resources such as patient data, clinical notes, guidelines, and online publications play a vital role in medical knowledge base evolution, but the most important resource is physicians' heuristics, experiences, and practices for knowledge creation and its validation. On the other hand, physicians have less time to create/evolve the knowledge due to their hard schedule in practices. Therefore, the knowledge dissemination, in the form of its shareability and interoperability aspects, is one of the requirement to CDSS adoption [15, 32, 33]. Existing systems either create non-shareable knowledge or have minimal support to the clinical standards in the knowledge creation. In literature, some of systems are able to create shareable knowledge using standard knowledge representation, but it is not interoperable and not able to directly integrate in diverse clinical workflows [34, 35].

Medical standards such as knowledge representation standards, data model standards, and terminology standards can achieve the knowledge shareability and interoperability on one hand, but it increases the knowledge creation complexity with respect to usability. Therefore, a flexible and robust model is needed to create shareable and interoperable knowledge with user-friendly environment. In literature, it is observed that the current knowledge acquisition methods either lack shareable knowledge creation or able to create shareable knowledge with high authoring complexity [36,37] and minimal interoperability support. The main goal of this work is to provide a flexible and robust mapping model to enabling a physician-friendly (user-friendly) knowledge authoring environment for shareable and interoperable knowledge creation more accurately. To achieve this stated goal, following are some challenges to face in this research work.

- How to design and develop a mapping model to ensure the accurate knowledge creation?
- How to provide multi-modal mapping among standard and non-standard data models and terminologies with remarkable accuracy to achieve the interoperability aspect?
- How to provide maximum abstraction for hiding the technology-oriented complexity from physicians and how to automate the knowledge transformation from non-standard to standard representation scheme?

We proposed a flexible and robust *semantic reconciliation model* (SRM) to create shareable and interoperable knowledge with high level of abstraction to physicians for hiding technological complexity. SRM provides following two solutions to achieve the aforementioned goal and to address the above challenges.

- **Multi-model Mapping:** The rules (knowledge base) execution usually depends on the schema and data of legacy systems. Therefore, SRM provides *Schema-Data Level Semantic Reconciliation* to map standard and non-standard terminologies (i.e. SNOMED CT), data models (i.e. vMR), and localized ontology (i.e. domain clinical model DCM). It maps standard data model to standard terminology, standard terminology to DCM, and DCM to standard data model. A significant mapping accuracy is needed for achieving interoperability.

- **Automatic Shareable Knowledge Generation:** The structure and syntax of shareable knowledge representation is hard to follow by physicians. Addition to structural and syntactic complexity, the amalgamation of standards to knowledge further increase the complexity level. SRM also comprises *Structure Level Semantic Reconciliation* to map the knowledge rules into standard representation structure and syntax. It transforms the created rules into standard representation with convergence of standard concepts, which are outcome of the multi-model mappings.

1.4 Proposed Solution Overview

We propose a *semantic reconciliation model* (SRM), which overcomes the limitations of existing systems regarding knowledge acquisition along with interoperability and shareability aspects of knowledge. The knowledge shareability is achieved with standard knowledge representation such as HL7 Arden Syntax-based Medical Logic Module (MLM). SRM normalizes the knowledge acquisition complexity with utilization of localized concepts DCM, and achieves the interoperability with the help of standard data model (i.e. vMR) and standard terminology (i.e. SNOMED CT). We converge SNOMED CT codes into the MLMs for avoiding the intrinsic vocabulary binding issue of Arden Syntax [4, 38]. According to our experience with physicians of our collaborative hospital, Shaukat Khanum Memorial Cancer Hospital and Research Centre (SKMCH)[1], we incorporated hospital management and information systems (HMIS) concepts in the form of the *domain clinical model (DCM)*. Physicians are comfortable with localized concepts instead of standard concepts as a binding vocabulary with data model (i.e vMR) [18, 39, 40]. The *semantic reconciliation model* (SRM) is realized through our designed and developed user-friendly authoring environment, called *Intelligent-Knowledge Authoring Tool* (I-KAT), which is deployed in our collaborative hospital.

The proposed SRM provides a flexible and robust multi-model mapping algorithms to map standard and non-standard terminologies, data models, and localized concepts. The multi-model mapping algorithm *Schema-Data Level Semantic Reconciliation* deals with two types of mappings, a) mapping between non-standard concepts such as DCM and standard terminology SNOMED CT, b) mapping between standard data model such as vMR and standard terminology such as SNOMED CT. In first case, the concepts mapping achieve with high accuracy by embedding explicit semantics, while in second case the standard data model and terminologies are usually mapped based on their concepts definitions. The resultant mappings of a) DCM and standard terminology and b) standard terminology and standard data model, are used to map DCM and standard data model by transitivity law.

SRM transforms the created rules into shareable format of standard knowledge representation (i.e. MLM) using S*tructure Level Semantic Reconciliation* algorithm. It also replaces the local concepts of DCM, used in created rule, with corresponding mapped standard concepts of data model and terminology for easy integration of the knowledge base with legacy systems. The resultant knowledge base can easily disseminate and integrate in healthcare institutions and clinical workflows. In this thesis, we targeted Head and Neck Cancer domain as a case study, but the system methodology is flexible to support other domains. The DCM is needed to change for the required domain, while the rest of methodology will be work in similar manner.

It is observed that multi-model mappings with high accuracy is a prerequisite for the shareable knowledge and it maximizes the interoperability of knowledge. Therefore, we improved the accuracy of the our proposed system by *Schema-Data Level Semantic Reconciliation*. The precision

[1]SKMCH: https://www.shaukatkhanum.org.pk/

of our system is 0.95, which is greater than the existing mapping algorithms: AgreementMaker Light (0.83) [41], LogMap Light (0.79) [42], and GOMMA (0.71) [43]. Usually, the ontology matching algorithms lack definition based algorithm, therefore, we evaluated our proposed definition based algorithm with base-line (Jaccard Similarity) algorithm for *Standard Terminology and Data Model Mapping*. As compared to base-line (Jaccard Similarity) algorithm, our proposed algorithm showed better results as precision (0.89), recall (0.97), and F-measure (0.93).

After realization of SRM using I-KAT, we evaluated the results of our system through system-centric and user-centric aspects. We compared the performance of our system with the ArdenSuite tool of Medexter [44, 45]. In the system-centric evaluation, we used state-of-the-art recommendations [24] and observed that our system exhibited higher implementation support for all three requirement categories (i.e., *Essential*, *Recommended*, and *Optional*) [24]. In the user-centric evaluation, we focused on the ease of use and reduction of MLM syntactic and logical errors. In the evaluation process, physicians and knowledge engineers were asked to create MLMs for treatment plan recommendations for an oral cavity site in the domain of head and neck cancer. On average, our system improved participant performance by a factor of 15 over Arden-Suite. The average error rate for MLM creation using our system decreased from 13 errors to just one error. We also validated our system on 1314 real cases of SKMCH oral cavity patients and verified the execution environment with the created knowledge base. We also evaluated the proposed system efficiency with respect to time on task and task success rate. The task success rate of the proposed system was 90.625%, while the existing system was 46.87%. Based on the ratio of mean time completion to the success rate, the overall efficiency of the proposed system was 56.62%, which better than the existing system efficiency 1.85% completion rate/time.

1.5 Contributions

The goal of this research work is providing a *semantic reconciliation model* (SRM) to create shareable and interoperable knowledge with high level of abstraction on technology-oriented complexity. Based on this comprehensive solution, we have two main objectives, multi-model mappings with high accuracy, and easy creation of shareable and interoperable knowledge. To achieve these objectives, we have following contributions in this research work.

Multi-model Mapping: Implicit Semantics Inset

Use of standard terminologies in a knowledge base maximizes the interoperability feature. On the other hand, physicians prefer to use localized concepts instead of standard terminology. Therefore, an accurate mapping is needed between standard and non-standard (localized) ontologies. The SRM provides multi-model mapping among local ontologies, standard terminologies, and standard data models. Addition to the existing mapping algorithms, we embed insights of the concepts using *Implicit Semantic Inset* and *Explicit Semantics Inset* to increase the mapping accuracy. While the existing mapping algorithms only focus on the internal structure of the ontologies.

Multi-model Mapping: Definition-based Mapping

In Multi-model mapping, the *Schema-Data Level Semantic Reconciliation* also focuses on the standard data model and standard terminologies mapping. Usually, standard data models have less number of classes and attributes than the number of standard terminologies concepts. Concepts in some particular hierarchy of standard terminologies are binded with some specific attribute of a class in the data model. The binding of attribute of data model to a specific hierarchy concepts can only be mapped by the definition and purpose text of attribute and hierarchy of terminology. This type of mapping is difficult with existing mapping algorithms, which are focusing on string matching, label matching, structure matching, child matching, path matching, and many others. The existing mapping algorithms lack definition-based mapping. Therefore, we propose the definition-based mapping algorithm, which maps the definition text of data model attributes with definition of hierarchical concepts of standard terminologies. In definition-based mapping, we recognize the insight of text using concepts with their implicit and explicit semantics in definition.

Automatic Generation of Shareable Knowledge

According to our goal of shareable and interoperable knowledge generation, the SRM is equipped with *Structure Level Semantic Reconciliation* algorithm to transform the plain rules of knowledge base into shareable knowledge representation. This algorithm provides the high level abstraction on the technological aspects such as structure and syntax of the standard representation. The existing systems of shareable knowledge creation provide interfaces to physicians without abstraction, and physicians are responsible to deal with structure and syntax of the standard representation. Additionally, our algorithm also converge the standard data model and terminologies concepts into the create plain rules with respect to use local concepts. This convergence takes help from the mapping files generated in multi-model mapping phase.

System-centric Evaluation Based on Requirements Completeness

In this research, we introduce requirement completeness evaluation under the system-centric evaluation criteria. In this evaluation, we compare our developed knowledge authoring tool I-KAT based on SRM realization, with benchmark system for the requirements clinical information modeling tool developed by Moreno-Conde et al. in [24]. We propose a four phase model to refine the requirements list, which are specifically deals with knowledge acquisition tools. The requirements are divided into three categories such as essential, recommended, and optional. In the result of four phase model, total 39 out of 56 requirements are selected as the knowledge acquisition tool requirements. These requirements can evaluate and compare with benchmark systems.

1.6 Thesis Organization

This dissertation is organized into chapters as following.

- **Chapter 1: Introduction**. Chapter 1 provides brief introduction of the research work on semantic reconciliation model for medical systems and in particular the role of semantic reconciliations in achieving knowledge shareability and interoperability among CDSS and HMIS systems. It focuses on the problems in the areas, the goals to achieve these problems, and finally the objectives achieved in this research work.

- **Chapter 2: Related Work**. A background detail is provided in this chapter about the multi-model mapping using different matching techniques and approaches, for achieving semantic knowledge interoperability. This chapter also provides the state-of-the-art literature for the knowledge shareability aspect. Finally, it provides comparison of these systems with the proposed system of the research thesis to reflect the limitations of current systems addressed by the proposed system.

- **Chapter 3: Proposed Methodology: Semantic Reconciliation Model**. A proposed solution in the form of a framework for achieving knowledge shareability and interoperability is presented in this chapter to overcome the limitations of current approaches. This chapter also provides overview of the concepts used in the thesis related to the proposed approach. It defines the scope of the thesis in achieving the knowledge shareability and interoperability among CDSS systems and medical systems such as HMIS.

- **Chapter 4: SRM: Schema-Data Level Semantic Reconciliation**. A semantic reconciliation model for multi-modal mapping is presented in this chapter that is used for generating the mappings between localized concepts (domain clincial model DCM) and standard terminologies. The second part of this chapter provides the methodology for mappings among standard data model and standard terminologies. Finally, it represents the methodology for DCM and standard data model mappings.

- **Chapter 5: SRM: Structure Level Semantic Reconciliation**. This chapter will explain about *Structure Level Semantic Reconciliation* to transform the production rules into shareable and interoperable knowledge. The shareability aspect of the knowledge is achieved by standard representation of Knowledge such as Medical Logic Module, while the interoperability is achieved with the help of our previous solution of *Schema-Data Level Semantic Reconciliation*, discussed in previous chapter..

- **Chapter 6: Results and Evaluation**. The results and evaluation of different techniques used in the proposed framework are highlighted in this chapter. It explains two types of results and evaluation. Firstly, it describes the results of multi-model mapping and evaluates with existing systems. Secondly, the automatic shareable knowledge creation system is tested and evaluated with existing knowledge authoring systems.

- **Chapter 7: Conclusion and Future Directions**. This chapter concludes the thesis and also provides future directions in this research area. The main contribution of the thesis is also highlighted in this chapter.

Chapter 2

Related Work

Knowledge shareability and interoperability demands clinical standards of knowledge representations, data models, and terminologies. The needs of interoperability such as multi-modal mappings among standard and non-standard terminologies and data models, are prerequisite to make the knowledge shareable and interoperable. Therefore, we divided the related works in two sections, Knowledge Interoperability: Multi-model mappings, and Knowledge Shareability: Shareable knowledge approaches.

2.1 Knowledge Interoperability: Multi-model Mappings

Knowledge interoperability requires the mapping among different standard and non-standard terminologies, but the existing ontology matching techniques focus on the standard ontologies only. Matching techniques generate mappings, which are contributing in interoperability among different systems. In literature, many ontology matching techniques are utilized for generating mappings among different ontologies. In this section, we will review existing ontology matching techniques without domain restrictions.

We focused on some of ontology matching techniques based on their participation in Ontology Alignment Evaluation Initiative (OAEI) [46], and some of the state of the art systems. Falcon-AO [47] ontology matching system has achieved and shown best results in the first few years of OAEI competition [48]. It provides algorithms to match, align, and learn ontologies using divide and conquer approach on large medical and non medical ontologies and gives 1:n alignments as output [49]. This system is very effective to align large ontologies by providing matching techniques along with user interfaces, however, it lacks extendibility and reusability features in the system. Falcon-AO++ [50] is the extended version of Falcon-AO, which provides a methodology for a domain expert to interact with system during alignment process. The domain experts intervention provides a slight improvement with respect to performance, but it makes the system highly dependent on the domain expert's input. Therefore, the system is difficult to extend for adding new matching algorithms to cater with large biomedical ontologies.

GOMMA [43] provides a methodology to manage, match, and evolve different healthcare ontologies. In year 2012, GOMMA was got second position in OAEI ontology matching competition and was able to map very large life science ontologies with high accuracy and runtime performance [51]. A group of researchers have introduced an ontology mediation approach, Semantic Information Layer (SIL), to achieve data interoperability among Enterprise Information Systems (EIS) [52]. SIL provides mapping services by extracting data from different data sources, querying the mapping information, and providing the information to the upper layers.

The extendibility issue in ontology mapping frameworks is resolved by Agreement Maker [53]. The system provided a visualization environment to support several mapping layers visually, and it automatically presents the generated mapping for further alignment. This system provides very extensible and flexible platform using comprehensive interfaces, but it lacks the scalability feature to match large ontologies. In extension of Agreement maker, the authors resolved the scalability issue with new framework AgreementMakerLight (AML) [41]. AML followed and preserved the methodology of Agreement Maker framework and enhanced computational efficiency to deal with large scale ontologies matching. AML competed with state-of-the-art matching systems and positioned as a top performer in the recent OAEI competition [46]. The system YAM++ extended the generic version of YAM-BIO to increase ontology matching efficiency [54]. The system provides a disk-based ontology matching methodology by utilizing indexes to achieve high efficiency in large scale ontologies, and also supports flexibility, extensibility, and self-configuration features to combine individual matchers [55]. The authors have used information retrieval algorithms to find mappings among ontological concepts, and also focused on problems of multi-lingual ontologies mapping algorithms [56].

In current literature, another ontology matching tool LogMap also deals with scalability issue in large ontologies matching [57]. It produces very well-ordered and clean set of results for the generated mappings. In OAEI 2017 competition, LogMapLite participated with higher performance than LogMap, with respect to the shortest runtime and with high precision and recall [58]. However, the speed performance of LogMapLite is a direct influence of not utilizing reasoning, repair, or semantic indexing of LogMap [42]. A research group introduced XMAP, which establishes similarity between linguistic and structural context and have used external knowledge resources with the help of domain experts [59]. The domain experts are needed to interact with system to filter out candidate mappings. The biomedical versions of LogMap and YAM++ are introduced as LogMap-Bio and YAM-Bio, respectively [60, 61]. LogMap-Bio mapped top 10 most suitable ontologies, which are retrieved from the external oracle [62]. On the other hand, YAM-Bio utilized a predefined background knowledge [61], including mappings from DOID and UBERON ontologies to Foundational Model of Anatomy (FMA) [63], National Cancer Institue Thesaurus (NCI) [64], and SNOMED CT [29].

The POMap system focused on syntactic matching algorithm to compute similarity between two nodes to generate 1-1 mapping pairs [65]. It utilized the semantic matcher with external source Uberon, it also used structural matcher based on classes and subclasses. However, the POMap system is new participant in field of ontology matching and it lacks scalability feature and property-based matching techniques [66]. Similarly, SANOM has been proposed with probabilistic approach for estimating the optimal solution for matching process [67]. A advanced version of WikiMatch [68] is introduced, called WikiV3 [69], to search and match the ontological concept with Wikipedia and the matching results are used as an external source in different matching algorithms.

Table 2.1 summarizes and compares our proposed matching methodology with some of existing systems with respect to features of *Mapping Schemes*, *Semantic Mapping*, *Flexibility in Mapping Representation*, *Definition-based Mapping*, and *Accuracy of Mappings*. *Mapping Schemes* attribute indicates whether the system has its own matching algorithms or using some existing. The feature *Semantic Mapping* shows the system capability to match the ontologies using semantics matching. Similarly, the feature *Flexibility in Mapping Representation* mentions about the flexibility of system with respect to mapping representation. *Definition-based Mapping* feature shows whether the algorithm contains the matching of concepts with respect to their definitions. While the attribute *Accuracy of Mappings* shows the system is focusing on accuracy level instead

Table 2.1: Features comparison among state-of-the-art mapping systems.

Systems	Mapping Schemes	Semantic Mapping	Flexibility in Mapping Representation	Definition-based Mapping	Accuracy of Mapping
GOMMA	✓	✗	✗	✗	✓
LogMapLight	✓	✗	✓	✗	✓
AgreementMakerLight	✓	✓	✓	✗	✓
Falcone-AO++	✓	✗	✗	✗	✗
YAM++	✓	✗	✓	✗	✓
POMap	✓	✗	✗	✗	✗
Proposed System	✓	✓	✓	✓	✓

of performance.

2.2 Knowledge Shareability: Shareable Knowledge Approaches

The standard representation of knowledge is the leading source of knowledge dissemination among diverse medical institutions. In literature, most of the systems focus on generic standard representation of knowledge instead of medical standard representation. Fortunately, medical domain provides several standards specific to knowledge representation such as Medical Logic Module (MLM) [20], Guideline Interchange Format (GLIF) [70], Shareable Active Guideline Environment (SAGE) [71], and many others.

In this section, we describe existing knowledge authoring methodologies with standard and non-standard knowledge representations. As a instance of knowledge representation, we focus on the well-known knowledge representation HL7 Arden Syntax Medical Logic Module (MLM). In the existing Arden Syntax-based CDSS systems, MLMs are generated either manually or semi-automatically. Samwald, M. et al. [44] used MLMs generated by a commercialized tool in an integrated Arden Syntax development and test environment (IDE) in the CDSS system. Their MLM development tool provides a simple interface to create and test Arden Syntax MLMs. They are also working on solve the curly brace problem with the help of GELLO (a loose acronym for Guideline Expression Language Object-Oriented) and vMR standards. However, physicians are responsible for creating the MLM using its structural slots and need to be familiar with the MLM's syntax and structure, which is a tedious task. Therefore, domain experts greatly depend on medical informatics experts to transform knowledge into knowledge base. Similarly, there is no automation in the utilization of standard terminologies during MLM creation; physicians use standard concepts based on their own knowledge about standard terminologies.

A number of commercial vendors such as Agfa, AllScript, McKesson, Medexter, and Siemens incorporated Arden Syntax in CDSS integrated health information system softwares [72]. The initial versions of Arden Syntax provided easy to use environment to experts, but its evolution made the syntax difficult to ready, understand, and compose. This resulted in organizations' need of hiring knowledge engineers to translate expert knowledge into Arden Syntax. Arden2ByteCode [73] is one such open source system that uses Eclipse framework for authoring environment to create and edit Arden Syntax MLM. Experts faced difficulty in understanding the complex environment of the system, therefore, they depended heavily on knowledge engineers to create knowledge.

Child Health Improvement through Computer Automated system (CHICA) [74] uses Arden Syntax MLM to generate encounter documents for patient visits. The system transformed the guidelines into algorithm to author MLM and required intense involvement of the knowledge engineers.

A. Soumeya et al. [75] proposed a rule editor for clinicians creating Arden Syntax MLMs, which has complex interfaces with multi-phase selection for the ontology domain; however, it lacks interoperability to integrate with external databases. Jung, C. Y. et al. [9] propose a system that executes MLMs represented in ArdenML. A third-party open source production rule system, Drool, has been used for MLMs execution. Therefore, the system needs two transformation phases: first, expert knowledge transformed into Arden Syntax; second, Arden Syntax transformed into ArdenML. In the second transformation, physicians need additional expertise in Extensible Markup Language (XML) and Extensible Stylesheet Language Transformations (XSLT).

Dustin et al. [76] proposed pattern and outcome-based approach to enable clinicians to create rules without dependency on knowledge engineers. However, this system was designed for the anesthesia domain only and is not scalable to other domains. Physicians also require training in XML language to create patterns and outcomes to utilize in rule creation. A. Seitinger et al. [77] implemented an Arden Syntax-based CDSS to generate guidelines for Lyme disease. This system uses fuzzy Arden Syntax and manually transforms text-based medical guidelines into Arden Syntax MLMs. Generating MLMs manually is very tedious and error-prone. In addition, this system does not take into account any standard clinical data models or terminologies in the Arden

Syntax MLM creation. The system proposed by Nathan C. Hulse et al. [78], are using the Clinical Document Architecture (CDA) as standard for knowledge creation. This standard enhances the complexity of the system for the reason that embedding the executable statements into CDA is preferable choice for sharing data not the rules. In [79], Sailors, R. Matthew have proposed a Microsoft Windows hosted MLM Writer to assist the knowledge engineer in creating Arden Syntax MLMs. The authors have developed easy to use interface, but creation of a single rule required multiple steps to perform. The developed system limits the type of MLMs, as it only creates the rules to simple threshold or time driven rules. The system lacks the controlled vocabulary in rule creation. The advanced users of MLM Writer are allowed to create MLM in free text and the system validates the syntax of created MLM. The user of the system must be familiar to Arden Syntax artifacts and syntax. In [38], the authors have improved their work of [80] regarding the bibliographic linkages and standardized database linkages. The system sends request to PubMed and retrieves the related articles to add the references using Arden Syntax Editor in appropriate slot of MLM. This system achieved the database linkages to some extent but still it requires the full interoperability feature to integrate with clinical organizations.

We describe most relevant systems to the knowledge authoring for decision support systems in Table 2.2. The comparison is examined with features of *Shareable Knowledge*, *Interoperability*, *Standard Terminology*, *User-Friendly*, and *Scope of System*. According to these features, four out of sex systems are used Arden Syntax MLM for knowledge shareability, while the interoperability feature with respect to standard data model is achieved only by two systems including our proposed system. The ArdenSuite [44] system's interoperability depends on the physicians' skills regarding the standard data model. Standard terminology also helps in interoperability maximization of the knowledge, therefore, we considered the standard terminology as a separate feature. The other two features user friendliness and scope of system show the ease of use with respect to user performance and extensibility of the system with respect to domain.

2.3 Comparison with Proposed System

In light of the aforementioned literature, CDSSs without knowledge acquisition tools are not adaptive in the real environment. In general, the existing systems evolve the knowledge bases by cre-

Table 2.2: Features comparison among state-of-the-art knowledge authoring systems.

Approaches	Shareable Knowledge (Arden Syntax MLM)	Interoperability (S. Data Model)	Standard Terminology	User-Friendly	Scope of System
KAT (UMLS-Based) [75]	✓	✗	✓	✗	Domain Specific
KAT (XML-Based) [78]	✗	✗	✗	✗	Extendable
Rule Editor [77]	✓	✓	✗	✗	Domain Specific
KAT (Anesthesia) [76]	✗	✗	✗	✓	Domain Specific
ArdenSuite [44]	✓	Depends on Physicians	✗	✗	Extendable
Proposed System (I-KAT) [81]	✓	✓	✓	✓	Extendable

ating MLMs manually or in a semi-automatic manner. However, those systems that automatically generate MLMs lack standard terminologies and standard data model practices, which hinders knowledge shareability and knowledge interoperability. The use of standard terminologies and data models may make a system difficult to use; therefore, the existing systems lack user-friendly interfaces for acquiring knowledge. The syntax and other artifacts of Arden Syntax are tedious and difficult for clinical experts to memorize, but the existing systems do not provide any facility to hide these complexities from the clinical experts. In addition, the existing knowledge acquisition tools focus on a specific medical domain to create knowledge instead of a scalable system design that can be easily extended to other domains. As a result, our proposed system provides a user-friendly authoring environment to create Arden Syntax MLM as shareable knowledge rules for intelligent decision-making by CDSS.

In summary, some factors are observed in the discussion that are considered as barriers in knowledge interoperability such as medical terminologies, healthcare standards, and matching algorithms. Therefore, high accuracy and precision is needed in the mapping system along with functionality of text based mapping. According to multi-model mapping systems, our methodology insets the implicit and explicit semantics into the mapping concepts and increases the accuracy and precision of the matching algorithm. Similarly, the current ontology matching systems only focus on string matching, label matching, child matching, and many others, however, the current systems lack the definition-based mapping of two ontologies.

According knowledge shareability, the existing systems generate shareable knowledge using manual or semi-automatic processes, which is tedious task for physicians. Therefore the proposed system generates the shareable knowledge (i.e. MLM) with providing high level abstraction, and hides the overall complexity of Arden Syntax MLM. The standard representation of MLM has a complex structure and syntax (i.e. Arden Syntax), it is difficult to remember all the syntax and structure by physicians. However, we hide this complexity and enhanced the physicians performance with respect to time and decrease the chance of errors during the knowledge creation.

2.4 Preliminaries

In this section, we describe the brief overview of the medical standards such as data model, terminologies, and knowledge representation.

2.4.1 Standard Data Model

Data, information, and knowledge are the central ingredients of healthcare systems. These constituents of autonomous organizations are represented in diverse format that leads to interoperability problem. Therefore, a uniform format and representation to communicate heterogeneous institutions. The heterogeneity in healthcare domain is at two levels: data and process [82]. All the processes can be handled with domain knowledge, therefore, the knowledge interoperability has worth for knowledge dissemination. Healthcare standards play an important role in achieving interoperability among different medical systems such as CDSS and EHR systems [83]. Standard data models such as HL7 Virtual Medical Record (vMR) [84] and HL7 Clinical Document Architecture (HL7 CDA) [85] plays important role towards interoperability among CDSS and Hospital Management and Information Systems (HMIS) [86]. In this research, the vMR standard data model is used as case study for interoperability. The vMR data model is based on the HL7 Reference Information Model (RIM) [87].

2.4.2 Standard Terminology

Standard terminologies are mainly utilized for standard encoding the clinical concepts used in the information and knowledge to disseminate with clinical communities with conflict in concept meanings. In literature, there a number of standard terminologies are used to achieve interoperability such as ICD10, SNOMED CT, UMLS, and FMA. In scope of this research thesis, we focused on the SNOMED CT. Systematized Nomenclature of Medicine – Clinical Terms (SNOMED CT) is multilingual standard vocabulary to exchange clinical information [88, 89]. International Health Terminology Standards Development Organization (IHTSDO) [90] is the international organization to manage all the versions of SNOMED CT.

2.4.3 Knowledge Representations

CDSS comprises of three main components such as knowledge base, inference engine, and user interfaces [91]. Smartness of CDSS is correlate with evolution of knowledge base. The knowledge base of a CDSS can be represented in a various formats depending on the domain and organizational preferences. Some of the knowledge representation are *Production Rules*, *Arden Syntax Medical Logic Module* (MLM) [79, 92], and *Guideline Interchange Format* (GLIF) [93]. In this thesis, we utilize the Arden Syntax MLM is a standard knowledge representation as an instance of our case study. HL7 have provided Arden Syntax language to represent clinical knowledge in order to facilitate the physicians for sharing their knowledge, practices, and experiences. Knowledge bases encoded in Arden Syntax are represented as a set of discrete and independent module called Medical Logic Modules (MLMs) [4].

2.4.4 Domain Clinical Model

The HMIS and electronic medical record (EMR) systems play a critical role in healthcare systems [94] to solve organizational problems to improve experts' performance and reduce the chance of errors [95]. Therefore, physicians are familiar with the localized concepts used in HMIS systems and EMR systems instead of standard clinical terminologies. A domain clinical model (DCM) is required to model all the used concepts in the local HMIS and EMR systems. The amalgamation of DCM into knowledge creation environment facilitates the physicians to create knowledge using understandable clinical concepts. Therefore, we modeled a DCM for our collaborative hospital using the local HMIS system of head and neck cancer.

Chapter 3

Proposed Methodology: Semantic Reconciliation Model

3.1 An Overview of Smart CDSS

Smart clinical decision support system (smart CDSS) assists the physicians to make decisions during the diagnosis, treatment, and follow-up phases in the patient care process. It provides recommendations to physicians and patients based on heterogeneous data sources including patients' clinical information, patients' status on social media, their daily life's behavior patterns, activities and emotions data [96]. Interoperability and shareability of data and knowledge among different HMIS systems and smart homes environment are considered as key challenges in healthcare domain. The knowledge shareability and interoperability depends on the data level and structure level mappings, which is the ability to communicate data among diverse HMIS systems [97] and to execute a unified knowledge for diverse input of data. These challenges can be resolved by resolving heterogeneity between different heterogeneous healthcare standards. The medical organizations, institutions, and hospitals, which are compliant to different healthcare standards, are the main consumers of Smart CDSS. Smart CDSS can only process information in vMR standard. Therefore, an adapter is needed to transform HMIS compliant healthcare standard to Smart CDSS compliant healthcare standard for the data level interoperability. However, system's knowledge base is also required to be executed with the standard input data. Therefore, Smart CDSS is equipped with a user-friendly knowledge authoring environment to evolve the knowledge base shown in Figure 3.1. In Figure 3.1, the **Authoring Tool** [18, 39, 81] facilitates the physicians to evolve the knowledge base of Smart CDSS. The Authoring Tool is realization of our SMR model to acquire and manage shareable and interoperable knowledge.

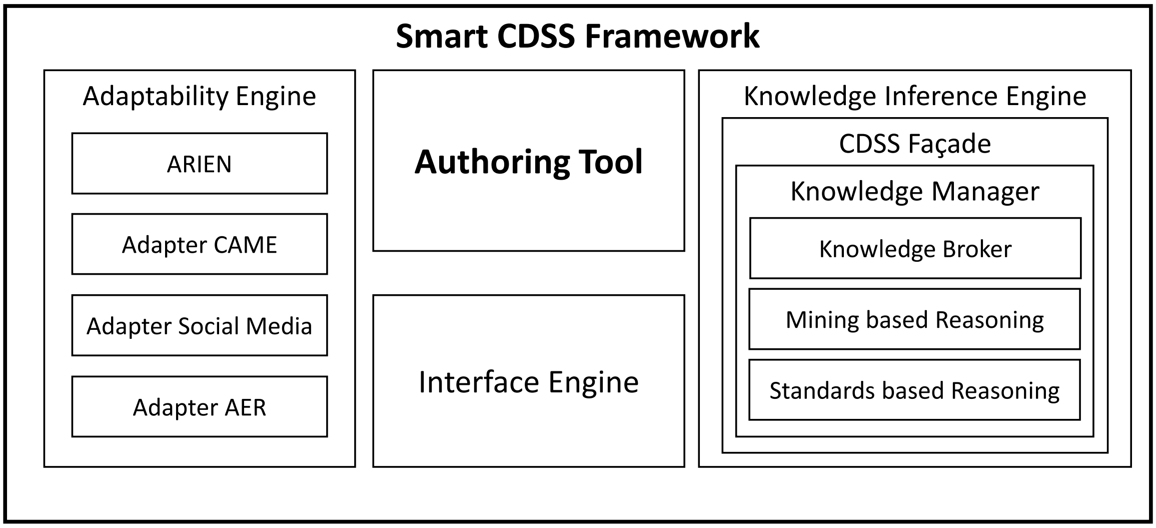

Figure 3.1: Architecture of Smart CDSS [1]

3.2 Knowledge Authoring Methodology

In the proposed system, we targeted head and neck cancer and focused on domain concepts for knowledge creation. Based on physicians' feedback regarding our previous work [18, 39], we encapsulated and provided abstraction of the vMR data model and the complex structure and syntax of the MLM from the user interface and provided the most usable and ordinary HMIS concepts for rule creation. The physicians feel easiness using local concepts instead of standard terminologies and data models. In proposed system, we used the localized concepts of HMIS in form of *domain clinical model* (DCM), it is considered as prerequisite for the system's user friendliness. The knowledge rules created in local concepts are transformed into standard data model and binding terminologies to make it interoperable. Therefore, a multi-model mapping is needed to map the DCM concepts to the corresponding standard data model classes' attributes and standard terminological concepts. According to shareability aspect, the created knowledge rules is transformed into standard representation of knowledge with convergence of multi-model mappings. In order to achieve the aforementioned goals, we proposed *semantic reconciliation model* (SRM), which provides a platform to multi-model mappings, called *schema-data level semantic reconciliation* and transformation platform to shareable knowledge such as standard MLM, called *structured* *level semantic reconciliation*. Additionally, the structure and standard of the shareable knowledge representation is also complex. The convergence of standard terminology and data model with knowledge representation increase the complexity level. In subsequent chapters 4 and 5, we will explain the *schema-data level semantic reconciliation* and *structured level semantic reconciliation*, respectively. Fig 3.2 illustrates the conceptual view of semantic reconciliation model (SRM).

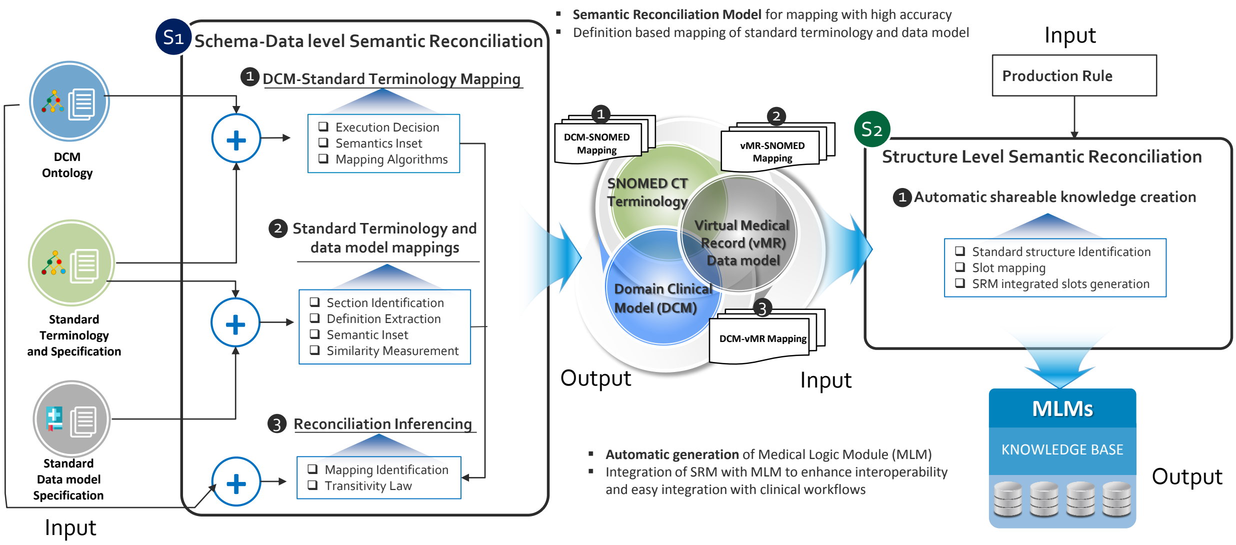

Figure 3.2: Conceptual diagram of Semantic Reconciliation Model (SRM).

In our research, we provided the abstraction by the *domain clinical model (DCM)* and the *semantics reconciliation model (SRM)* to handle the dependencies of HMIS, data model (i.e. vMR), and standard terminology (i.e. SNOMED CT) concepts. Therefore, we provided understandable DCM and SNOMED CT concepts on the user interface for rule writing, while shareable and interoperable MLMs were generated with standard vMR classes, attributes, and codes of SNOMED CT concepts in a back-end process.

3.3 Business Process Model and Architecture

The system's business process model for rule creation and mappings of DCM, vMR, and SNOMED CT terminologies is shown in Figure 3.3. Workflow is represented in standard Business Process Model and Notation (BPMN) format [98]. It resembles in domain analysis, formal notation selection, conceptual modeling, and logical modeling with the existing workflow process model [99]. The set of activities, processes, gateways, and messages is represented in pools with standard notations using Enterprise Architect [100]. High level system requirements in the form of the business process model are implemented and shown as system's architecture in Figure 3.4 . The system's workflow comprises two pools: *Physician Activity* and *Multi-Model Mapping*.

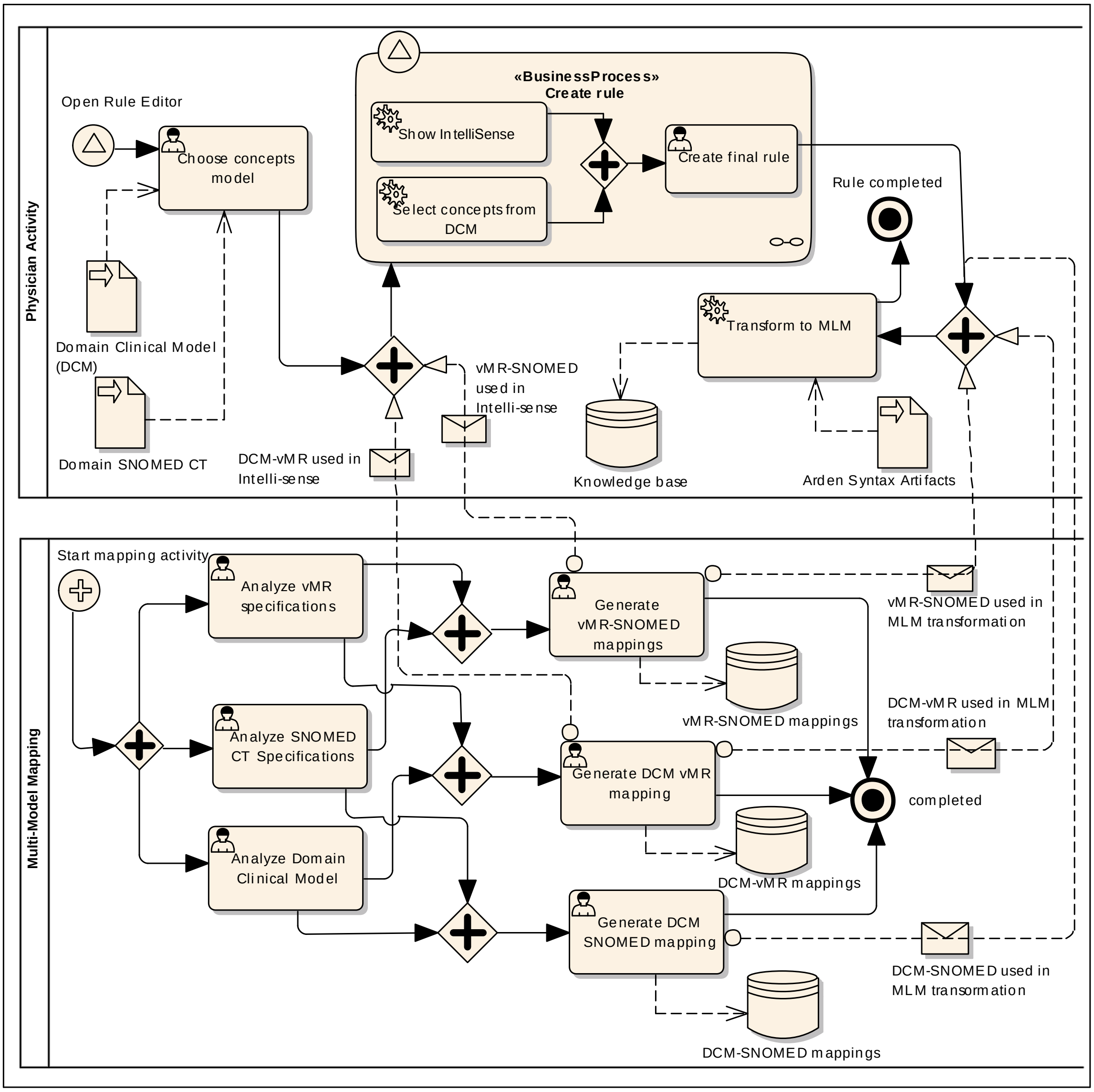

Figure 3.3: Rule creation and mapping generation and use in rule creation workflow.

3.3.1 Physician Activity Pool

The physician is the main actor in the system creating knowledge rules. In rule creation, the physician can use the desired concept values either from DCM or SNOMED CT terminologies. This enhances concept recall and reduces the chance of errors while enabling IntelliSense functionality. These concept models are selected by *Choose concepts model* from inputs of data model objects *Domain Clinical Model* and *Domain SNOMED CT*, respectively. The business process *Create rule* includes two parallel activities of concept selection (i.e., *Show IntelliSense* and *Select concepts from DCM*) to provide input for creating rule activity, *Create final rule*. The proposed system implements *Show IntelliSense* and emphSelect concepts from DCM on the *User Interface* module in architecture. In *Select concepts from DCM*, the physician can easily select the desired concepts for use in the rule facts. In architecture, these concepts are fetched to the *User Interface* using *DCM Concepts Controller* and *DCM Query Manager*, which creates and runs the appropriate query on DCM Ontology.

In *Show IntelliSense*, the physician is presented with a window that shows all possible values of the selected concept and allows selection of the correct desired value. The value list comes from the *Domain Ontology* using DCM-vMR and vMR-SNOMED mappings. The vMR schema classes and attributes bridge the selected DCM concept and the values list of SNOMED CT concepts. The *IntelliSense Controller* is the component responsible for performing *Show IntelliSense* activity using *DCM-vMR Mapper* and *vMR-SNOMED Mapper*. Both mappings are queried by three corresponding query managers (i.e., *DCM Query Manager*, *vMR Query Manager*, and *SNOMED Query Manager*). The final rule creation activity, *Create final rule*, is invoked using *Show IntelliSense* and *Select concepts from DCM* as parallel activities.

After successful *Create final rule*, the proposed system transforms the rule into Arden Syntax MLM by *Transform to MLM*. In summary, the rule is created by the physician using understandable DCM and SNOMED CT concepts, which the system then transforms into Arden Syntax MLM with amalgamation of vMR classes and attributes along with SNOMED CT codes. The *MLM Creator*, in system's architecture Figure 3.4, is responsible for performing *Transform to MLM*. The *Transform to MLM* involves three types of mappings generated by the *Multi-Model Mapping* pool using corresponding controllers and query managers. MLM has its own standard

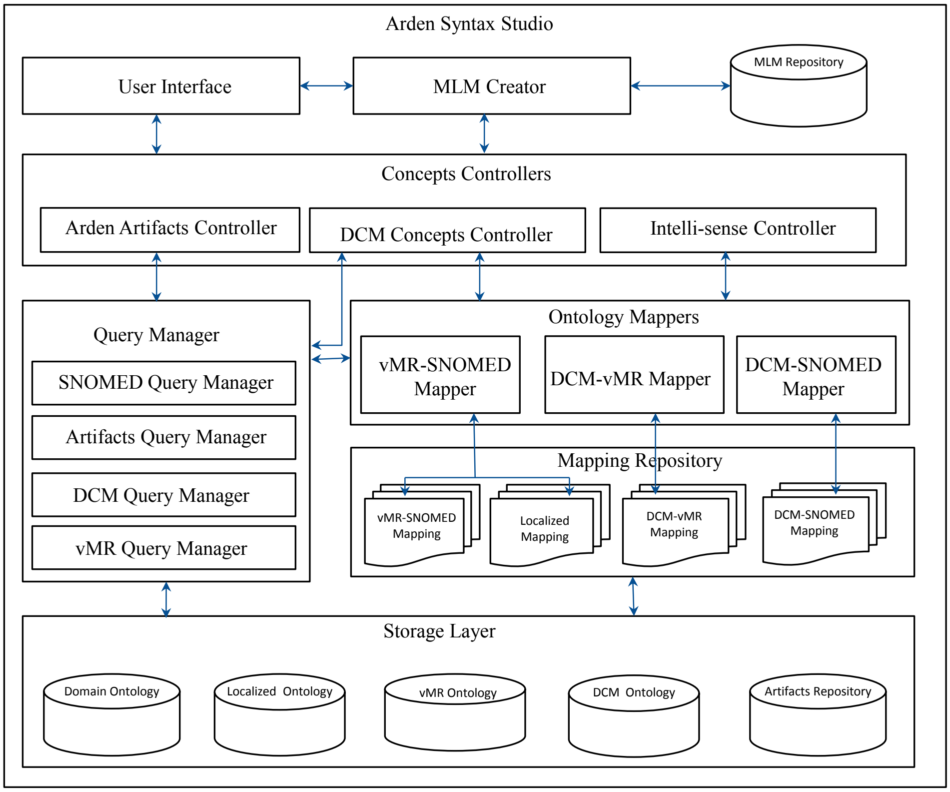

Figure 3.4: I-KAT architecture.

artifacts and syntax based on HL7 standard Arden Syntax specification. All artifacts are fetched by *Arden Artifacts Controller* using *Arden Query Manager*. The created MLM is stored in the knowledge base.

3.3.2 Multi-model Mapping Pool

In the business process model, the *Multi-model mapping* pool mainly focuses on generation of multi-model mappings among DCM, standard data model (i.e. vMR) schema, and standard terminology (i.e. SNOMED CT) concepts. This mapping activity is performed once as a prerequisite for rule creation. Three parallel activities of *Analyze vMR Specifications*, *Analyze SNOMED CT Specifications*, and *Analyze Domain Clinical Model* are performed to analyze vMR, SNOMED CT, and DCM, respectively. The outcome of vMR and SNOMED CT analysis is utilized by *Generate vMR-SNOMED mappings* to generate *vMR-SNOMED mappings*. Similarly, the outcome of vMR and domain clinical model analysis is utilized by *Generate DCM vMR mappings* to generate *DCM-vMR mappings*, while the output of SNOMED CT and DCM analysis is used by *Generate DCM SNOMED mappings* to generate *DCM-SNOMED mappings*. These mappings belong to our proposed *semantic reconciliation model* (SRM). The subsequent section 3.4 explains the *DCM* in detail. In our research, the multi-model mapping is handled in two ways, *a)* expert-driven multi-model mapping, explained in section 3.5 and *b)* schema-data level semantic reconciliation, explained in chapter 4. The former one is a semi-automatic process of mappings with involvement of physicians while the latter one methodology is fully automated without physicians involvement at the middle of process.

3.4 Domain Clinical Model (DCM)

The HMIS and electronic medical record (EMR) systems play a critical role in healthcare [94] to solve logistical organizational problems to improve experts' clinical decisions and reduce the cost of managing clinical information [95]. The HMIS and EMR systems are mostly used to maintain patients' active and inactive problems, allergy information, surgical, family and social histories, current medications, nicotine and alcohol use, symptoms, vital signs, and laboratory and radiology reports [101]. In general, physicians are familiar with these and other related clinical concepts. Therefore, the system facilitates creation of knowledge rules using understandable clinical concepts. The DCM provides a model to manage and organize the HMIS clinical concepts. We used a proper Clinical Information Modelling Process (CIMP) [24], based on investigating concept semantics. According to the standard requirements and recommended methods of the CIMP process, we collected clinical concepts from the HMIS system, analyzed and specified the clinical context among contents, and structured the DCM.

We structured the DCM concepts using a well-known and popular clinical notes protocol, SOAP (subjective, objective, assessment, plan). SOAP notes were developed by [102] to provide a logical and reproducible framework for generating medical records [103]. A SOAP-based model improves the quality of client services by easy communication among healthcare professionals and by enabling physicians to identify, prioritize, and track patients' problems in a timely and systematic manner [104]. Therefore, we designed a SOAP-based structure for DCM, which allows the clinical concepts to model in a scalable and manageable manner. We derived and aligned DCM with the HL7 standard data model vMR to maintain semantics among different concepts, as partially shown in Figure 3.5, as unified modeling language (UML) class diagram [105]. We transformed the DCM model into an ontology format, which was semantically verified by SKMCH physicians.

The information related to symptoms, past medical history, family history, social history, and current medication that exist in legacy systems are modeled under the *Subjective* category of the SOAP model. The *Objective* category includes vital sign and observable symptoms that can be easily measured through different physical tests, laboratory tests, and imaging tests. In the *Assessment* category, we organized all information about diagnoses, health status, and lifestyle changes of the patients. In the *Plan* category, we modeled all recommended treatment plans such as proposed medications, chemotherapy, radiotherapy, and surgeries. The individual DCM models are shown in Figure 3.6, 3.7, 3.8, and 3.9 for the *Subjective*, *Objective*, *Assessment*, and *Plan*, respectively, with attributes of the vMR data model.

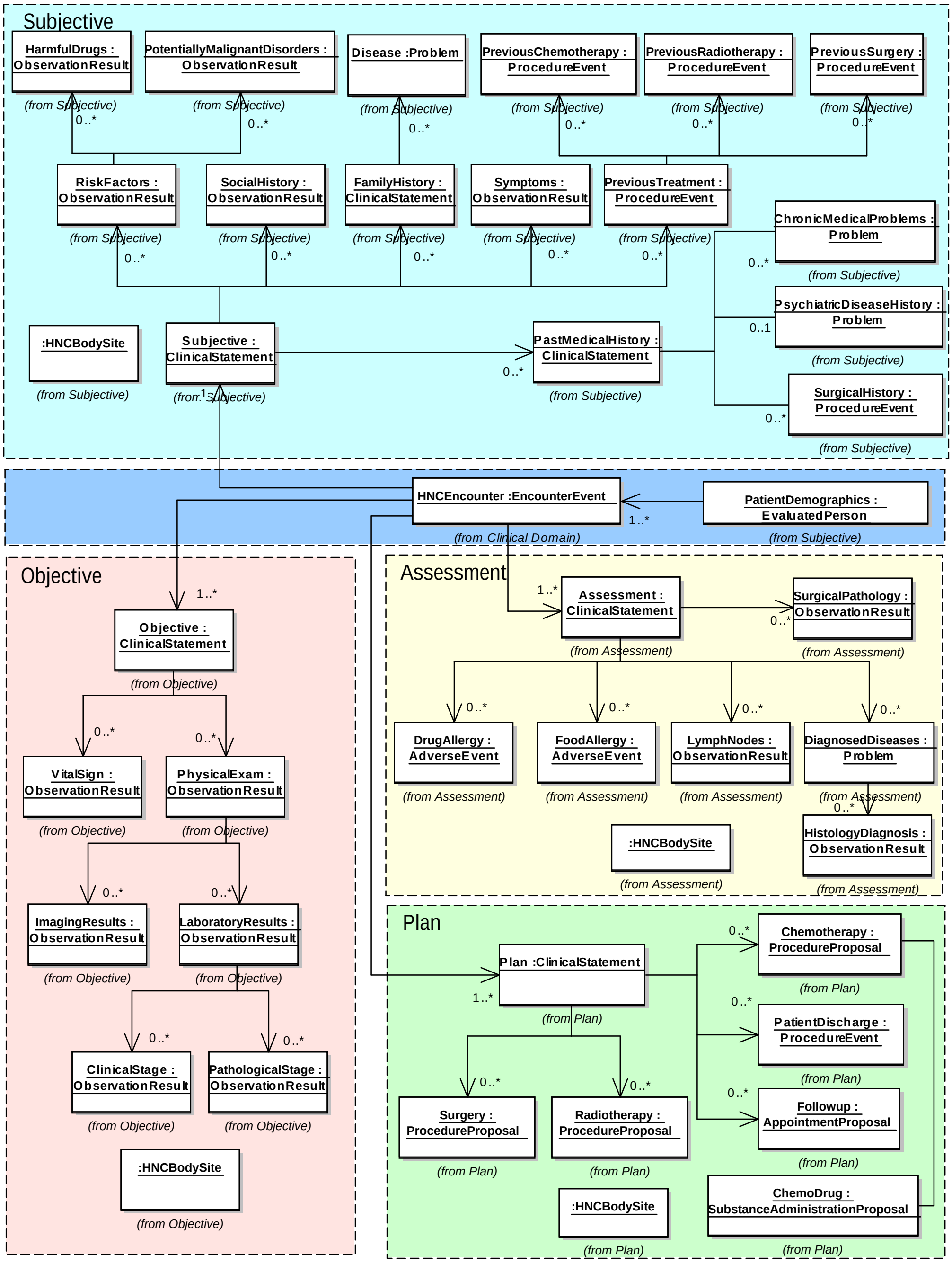

Figure 3.5: UML class diagram of domain clinical model (DCM) using a SOAP-based (Subjective, Objective, Assessment, Plan) protocol and vMR data model.

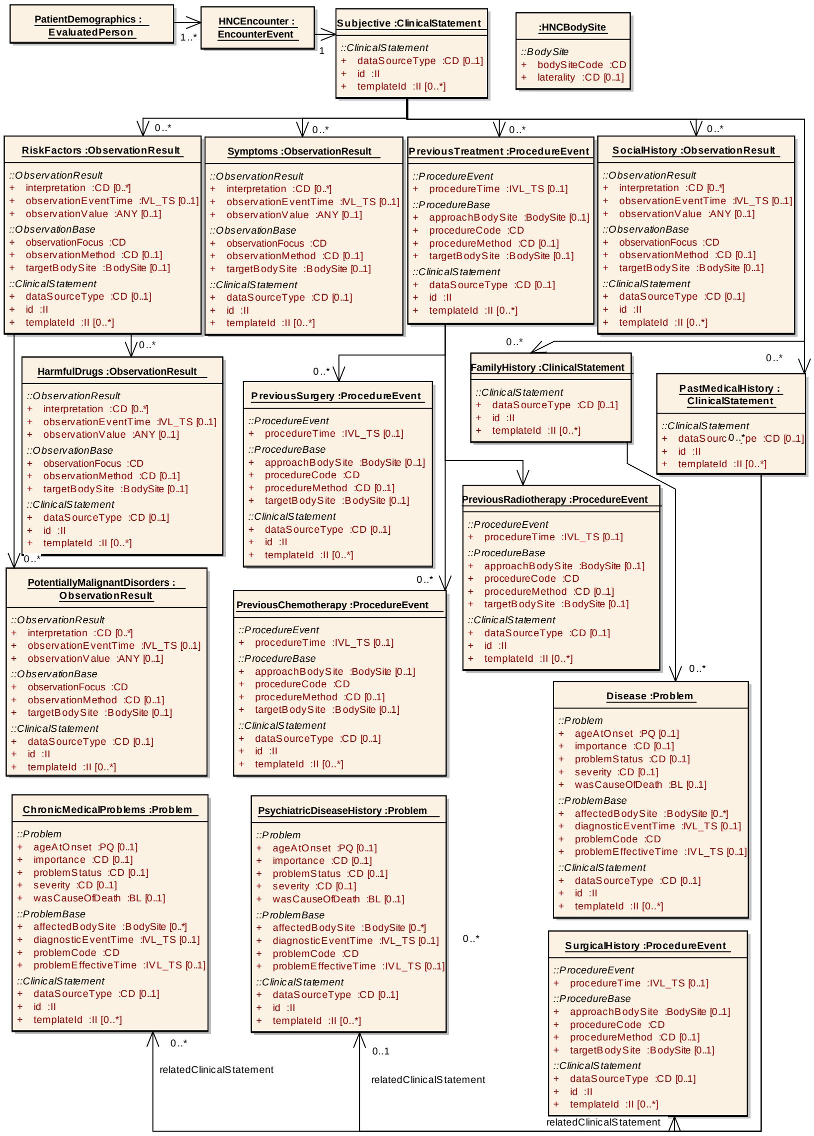

Figure 3.6: Subjective Model of DCM.

3.5 Expert-Driven Multi-Model Mapping

SRM is a reconciliation model that unifies concepts from three different models (i.e., SNOMED CT, vMR, and DCM) and reconciles it with high level abstraction. SRM achieves the objectives of interoperability, shareability, and user friendliness. While the system creates rules in MLMs using the standard vMR data model and SNOMED CT codes to achieve shareability and inter-

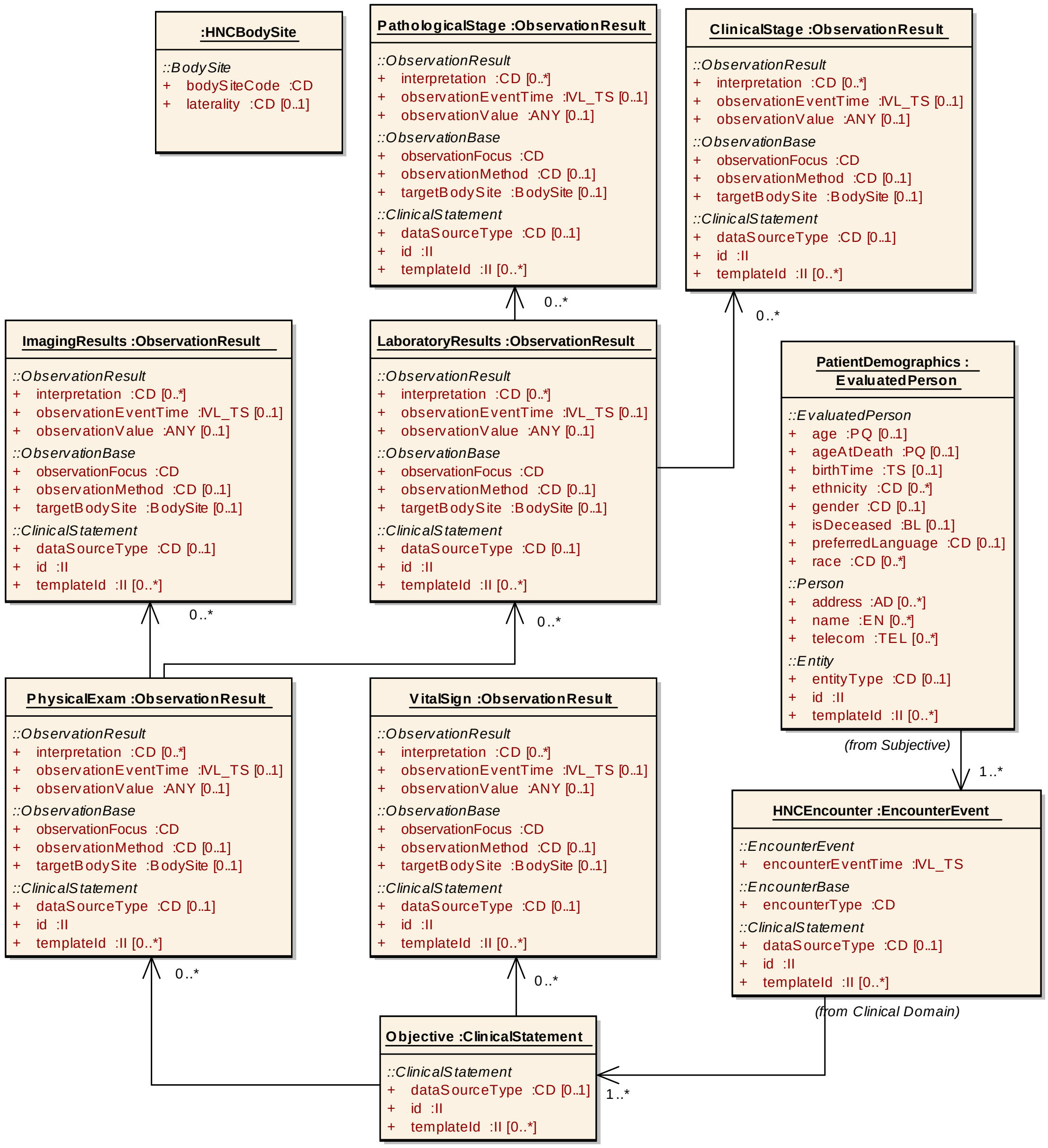

Figure 3.7: Objective Model of DCM.

operability, this increases the complexity of MLM creation for physicians. Therefore, the system hides this complexity and achieves user friendliness by providing a selectable tree of DCM concepts. It also provides SNOMED CT and DCM concepts in an IntelliSense window that allows physicians to select the desired concept. Consequently, the MLM's complex structure and syntax

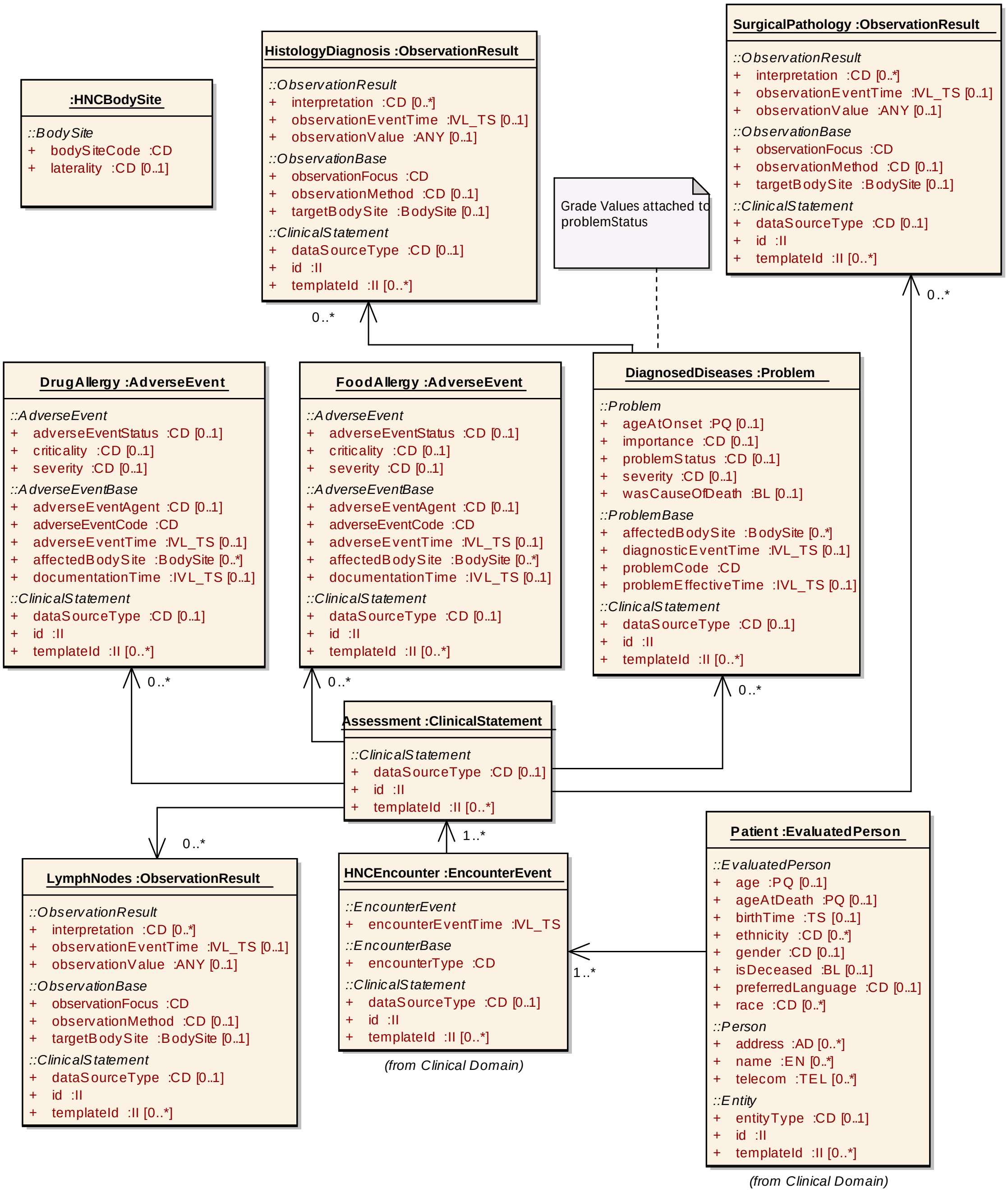

Figure 3.8: Assessment Model of DCM.

with consolidation of vMR and SNOMED CT are hidden from physicians. The SRM model, as shown in Figure 3.10, provides three types of mappings: DCM concepts to standard terminology

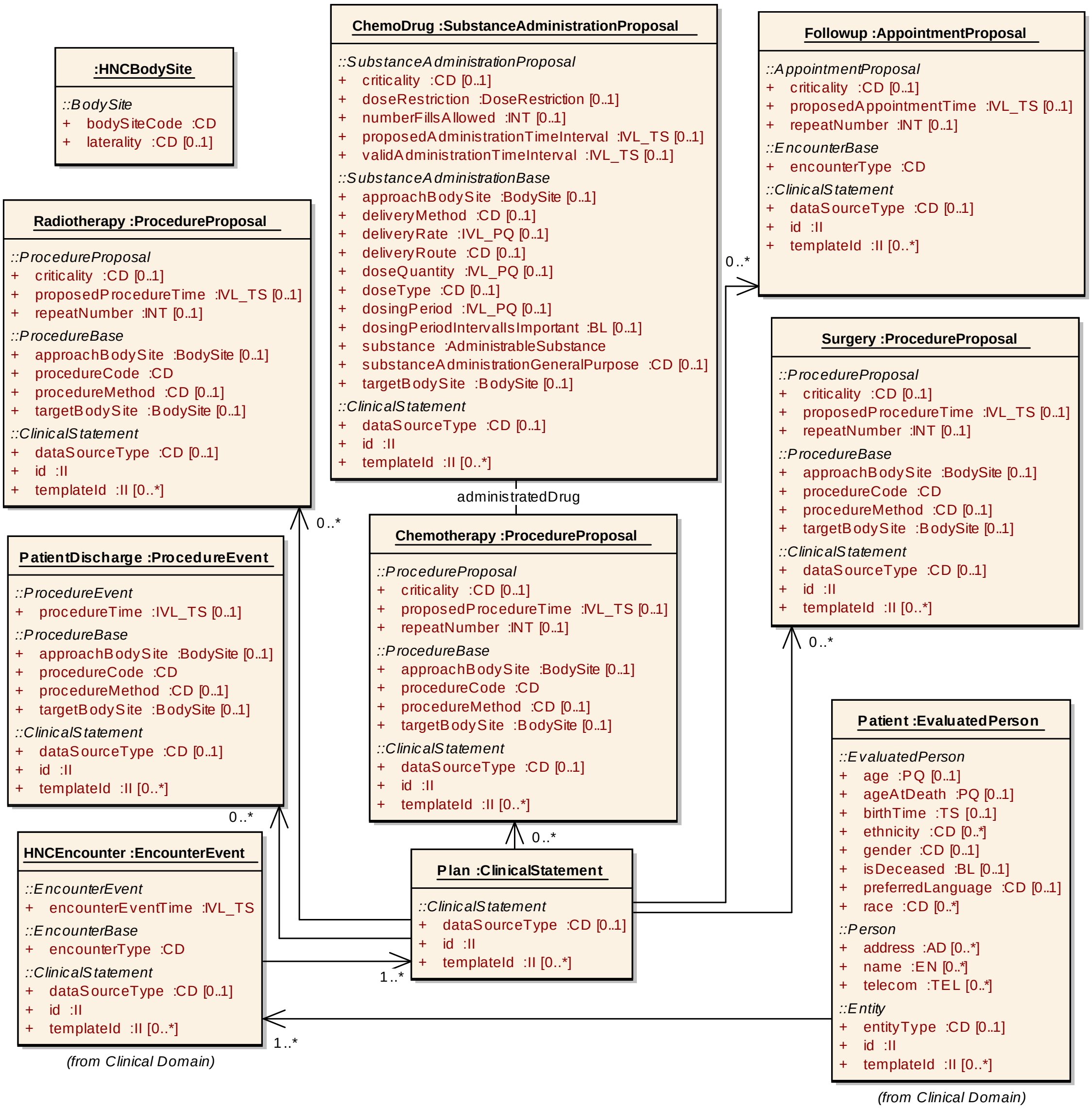

Figure 3.9: Plan Model of DCM.

i.e. SNOMED CT concepts (DCM-SNOMED), standard data model i.e. vMR to standard terminology i.e. SNOMED CT concepts (vMR-SNOMED), and DCM concepts to standard data model i.e. (DCM-vMR). These mappings are described in the subsequent sub-sections.

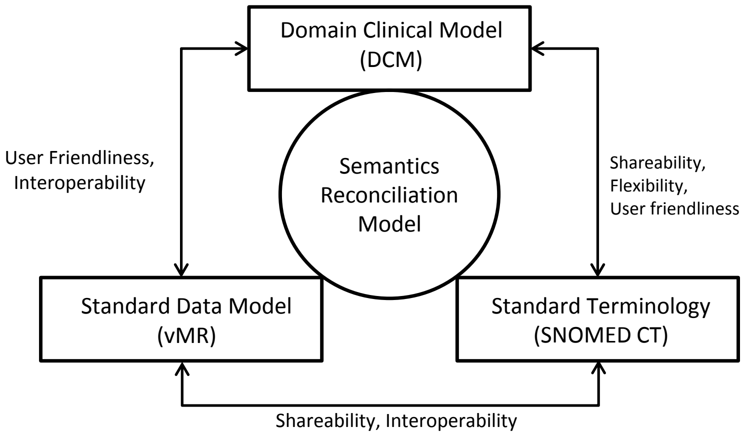

Figure 3.10: SRM: Multi-model mapping conceptual model.

3.5.1 Types of Mappings

DCM-SNOMED Mapping Using standard terminologies enhances the interoperability and shareability of knowledge acquisition tools [28]. In the SRM model, we mapped DCM concepts to SNOMED CT concepts to achieve shareability and user friendliness goals as physicians are more familiar with DCM concepts than vMR and SNOMED CT concepts. In the user interface, the physician selects the desired DCM concept during rule creation, while in the back-end process, the selected DCM concept is represented with vMR and SNOMED CT code in the automatically generated MLM. The system also provides physicians with choice of SNOMED CT concepts for achieving flexibility in the DCM concept selection.

vMR-SNOMED Mapping Standard data models and terminologies help in interoperable and shareable knowledge acquisition [26, 28]. These mappings are needed to achieve shareability and interoperability of the knowledge base. The system automatically transforms the created rule into Arden Syntax MLM with consolidation of the vMR and SNOMED CT codes of the correspondingly used DCM concepts in the rule.

Additionally, the vMR-SNOMED mapping helps concept selection in the IntelliSense window during rule creation. DCM concept scope may be compromised in some situations, requiring the physician to select “missing value”from the SNOMED CT. In this case, our system provides

IntelliSense functionality from SNOMED CT concepts instead of DCM concepts, which increases flexibility for concept selection. In these mappings, vMR schema classes and their attributes are mapped to the corresponding top hierarchy concepts of SNOMED CT. These mappings are verified by physicians and domain experts in the HL7 community.

DCM-vMR Mapping Understanding and memorizing all vMR schema classes and their attributes is a tedious task for physicians. Therefore, in the user interface, our system provides physicians with DCM concepts instead of vMR schema classes and attributes. These mappings offer user friendliness for knowledge creation and interoperability of the knowledge base. The DCM concepts are mapped to the corresponding vMR classes based on the DCM-SNOMED and vMR-SNOMED mapping output.

3.5.2 SRM Mapping Methodology: Example

In SRM, we focused on three types of mappings, i.e., DCM-SNOMED mappings, vMR-SNOMED mappings, and DCM-vMR mappings. In this semi-automated method, the DCM-SNOMED mappings are generated using our previously developed ontology matching system (SPHeRe) [82]. SPHeRe's matching algorithms include string, synonym, label, child, and property matching [25]. These algorithms are suitable for mapping SNOMED CT and DCM due to their ontological nature. We achieved 83.6% accurate mappings using SPHeRe. The remaining ambiguous and un-mapped concepts were mapped with the help of SKMCH physicians using the inspection method [106]. We extended our previous research with explicit semantic enrichment and mapped the DCM-SNOMED (DCM-Standard Terminology Mapping) with high precision and accuracy, which is described in chapter 4, section 4.1.

The vMR-SNOMED mappings were generated using the inspection method [106] involving different physicians. The vMR data model contains some specific and limited classes and attributes, which are mostly usable in CDSS systems. We selected the inspection method because vMR class attributes require coded values from particular SNOMED CT top-level hierarchical concepts. In the inspection method, the physician's role is essential because the vMR-SNOMED mappings depend on the semantics described in notes, definitions, descriptions, and purposes of each class as well as attributes of vMR and top-level hierarchical concepts of SNOMED CT. The methodology followed by physicians for mapping vMR schema classes to SNOMED CT top-level concepts is illustrated in Figure 3.11. The vMR-SNOMED mapping process contains two phases.

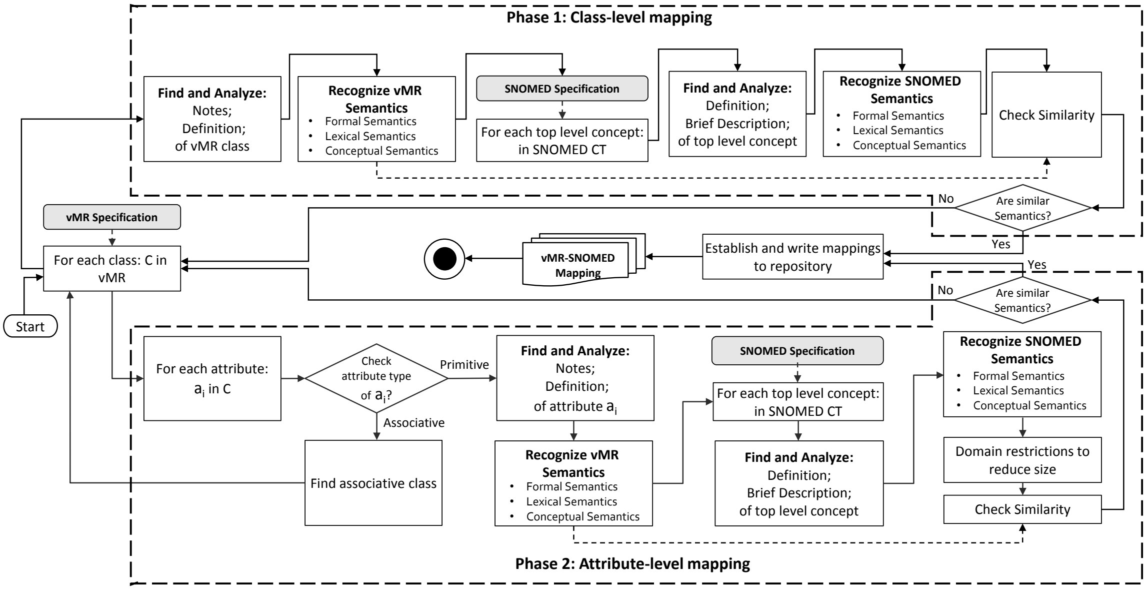

Figure 3.11: Expert-Driven vMR-SNOMED mapping process.

Phase 1: Class-level Mapping The vMR classes are mapped with corresponding top-level concepts in SNOMED CT. We used the vMR specification document *HL7 Virtual Medical Record for Clinical Decision Support (vMR-CDS) Logical Model, Release 2* [107] and SNOMED CT specification document *SNOMED CT Starter Guide* [108]. The vMR classes and attributes are described in the notes and definitions in its specification document. First, the notes and definitions of each vMR class are found and analyzed to understand the semantics of that particular class. Similarly, all SNOMED CT top-level hierarchical concepts are examined against the selected vMR class. The definition and short description parts of each SNOMED CT top-level concept are iteratively explored and analyzed. Physicians recognize the semantics (i.e., formal, lexical, and conceptual semantics) of the top-level concept. Based on the physicians' analyses of both the vMR class and the SNOMED CT top-level concept semantics, they compared the similarity and retained semantically matched concepts. For example, the vMR class *ProcedureEvent* is mapped to the SNOMED top-level concept *Procedure (71388002)*, as shown in Figure 4.8.

Phase 2: Attribute-level Mapping We mapped the attributes of vMR class with a SNOMED CT top-level hierarchical concept and its child concepts with specified domain constraints. Primitive attributes proceed with the normal mapping process; however, for associative attributes, the association class is first retrieved, and its attributes are mapped with SNOMED CT concepts using a recursion process. The associative attributes of the selected class are linked with its association class. Furthermore, the phase I steps for finding, analyzing, and identifying semantics are repeated. The additional phase II steps include finding attribute types, processing the association classes, and controlling the domain constraints on child concepts to reduce the size. Finally, vMR class attribute semantics are compared with SNOMED CT concept semantics to assess similarity. Similar semantics are considered as mapped concepts and are stored in the mapping repository.

The third mapping category is DCM-vMR mappings, which we achieved using the law of transitive relation, as represented by Equation 3.1.

$$\forall C_{DCM}, C_{snomed}, C_{vMR} \epsilon X : (C_{DCM} R C_{snomed} \wedge C_{snomed} R C_{vMR}) \Rightarrow C_{DCM} R C_{vMR} \quad (3.1)$$

We already mapped the DCM with standard terminology using *DCM-SNOMED mapping*, and standard data model vMR with standard terminology using *vMR-SNOMED mapping*, now the third one mapping DCM with standard data model vMR *DCM-vMR mapping* can easily achieve using transitivity law.

3.6 Summary

In the aforementioned multi-model mapping methodology, we focused on three types of mapping, DCM-SNOMED mapping, vMR-SNOMED mapping, and DCM-vMR mapping. As we mentioned that DCM-SNOMED mapping are achieved using our previous work of semi-automatic ontology matching system (SPHeRe) [82] with accuracy of 83.6%, and the remaining accuracy is achieved by involvement of physicians. Similarly, vMR-SNOMED mapping was achieved using expert-driven approach with full involvement of physicians. Therefore, we automated this process without involvement of physicians for DCM-SNOMED mapping, vMR-SNOMED mapping, and DCM-vMR mapping with our proposed *schema-data level semantic reconciliation* methodology, described in chapter 4. The verification and validation of mapping is performed by physicians after completion, while there is no need of physicians involvement in the middle of the mapping process.

Chapter 4

SRM: Schema-Data Level Semantic Reconciliation

The knowledge base of each decision support and recommendation system can be represented in different standard and non-standard representations such as production rules, medical logic modules, predicate logic, knowledge frames, and many others. The common facts in all these representations are condition and conclusion parts of the rule. Each rule has condition part and conclusion part, in further division each condition and conclusion has keys and values. The keys are usually dependent on the schema concepts of HMIS systems, while the values are the actual data instance of HMIS systems. For instance, a production rule ***IF disease = Head and Neck Cancer and Age*** $>=$ ***40 Then Treatment Plan = Surgery***, shown in Figure 4.1, contains the schema concepts (Keys) *Disease*, *Age*, and *Treatment Plan*, and the corresponding data instance concepts (Values) *Head and Neck Cancer*, *40*, and *Surgery*. The same rule is represented in two different HMIS systems HMIS-1 and HMIS-2 with different concepts, therefore, the same rule with different concepts can not be executed when it is not interoperable using some standard concepts.

According to the aforementioned example in Figure 4.1, we need schema and data level semantic reconciliation to deal with such type of problem. In knowledge interoperability, we focused on the schemal level interoperability and data level interoperability. Therefore, our proposed method *Schema-Data Level Semantic Reconciliation* provides multi-modal mappings among local concepts DCM, standard terminology concepts such as SNOMED CT, and for schema level standard data model such as vMR. Usually, the production rule is created in localized concepts of DCM, and it is required to transform into standard such as the schema concepts transformed into vMR concepts and data concepts transformed into standard SNOMED CT concepts code. The schema concept Disease is the ObservationFocus attribute of ObservationResult class, while the SNOMED

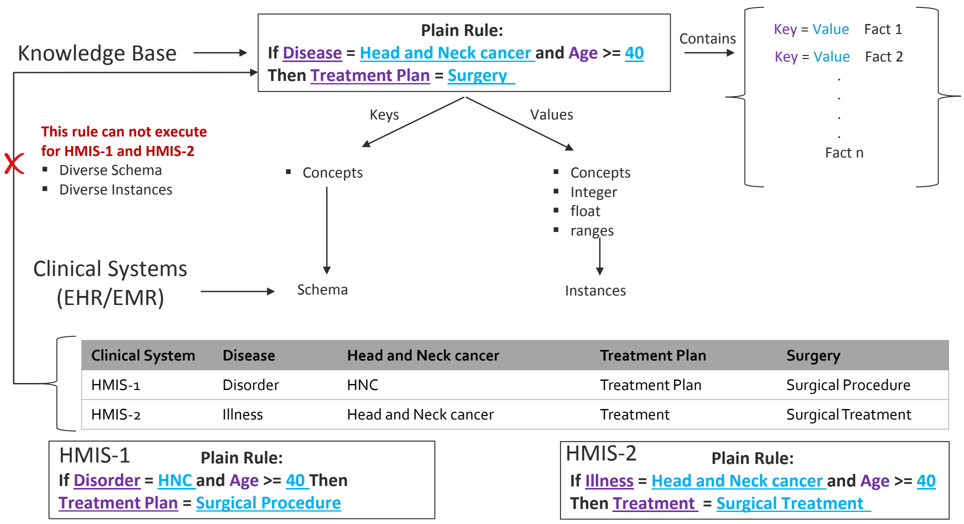

Figure 4.1: Example of rule representation in general

CT code of data concept Head and Neck Cancer is 255056009. The above production rule is transformed into standard format as follow and also shown in Figure 4.2 :

If ObservationResult.observationFocus = "64572001" and ObservationResult.ObservationValue = "255056009" and EvaluatedPerson.age >= 40 Then ProcedureEvent.procedureCode = "413737006" and ProcedureEvent.procedureMethod = "387713003". Using standard terminology concepts and data model class with attributes, the production rule can easily be integrated with legacy systems to execute the knowledge for diverse input of data. In order to achieve interoperability *Schema-Data Level Semantic Reconciliation* provides three types of mappings: DCM concepts to standard terminology (DCM-SNOMED CT), standard data model to standard terminology (vMR-SNOMED CT), and DCM concepts to standard data model (DCM-vMR), which is explained in the following sections. In previous chapter (Chapter 3), all these mapping were achieved using semi-automatic methodology with inspection method, now we designed and developed different algorithms to achieve these mappings with high accuracy and with less physician interventions.

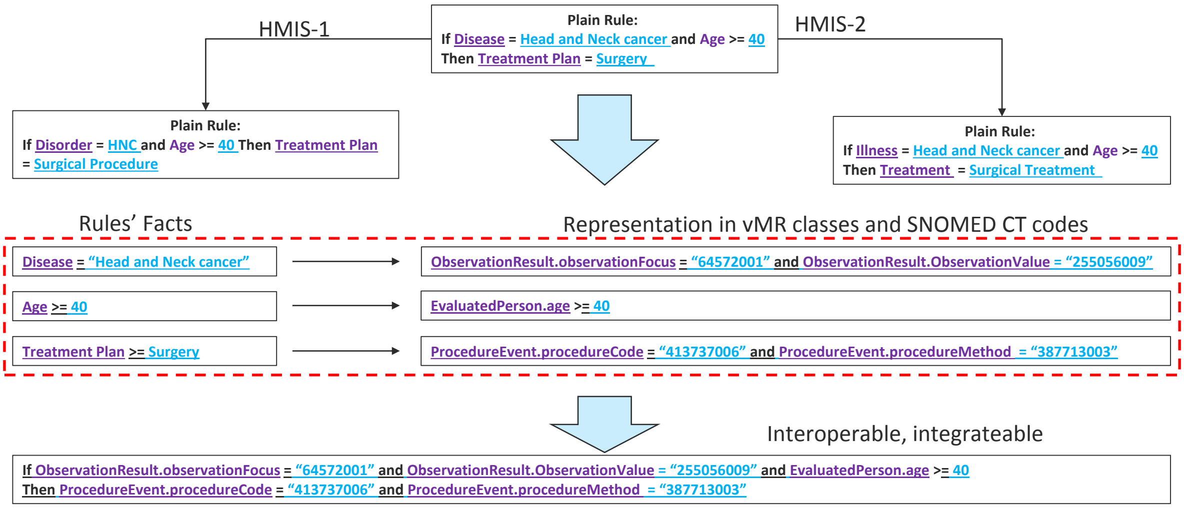

Figure 4.2: Example of rule representation in standard format

4.1 DCM-Standard Terminology Mapping

The current ontology mapping systems reflect the internal semantics of the source and target ontologies only and lack the external semantics during execution of mapping algorithms [47]. In the current context, the internal semantics mean that the relationships among the concepts of terminologies specified by the owner institutions based on their requirements, while the external semantics are the more generic relationships based on the standard terminologies. Therefore, in addition to the internal semantics, we inset the external semantics to the source and target concepts before the execution of matching algorithms. We transmute the short form of used acronyms, if exist, into their extended forms using acronyms library before the insertion of external semantics. The acronyms transmutation helps in enhancing the mapping accuracy up to some extent. In proposed mapping model, we inset the explicit semantics with the insertion of synonyms, hypernyms, hyponyms, and meronyms of the source and target concepts vectors. We used the existing two libraries, WordNet [109] and ConceptsNet5 [110], to inset the explicit semantics.

The proposed model allows to define the strategy for algorithms execution. The strategy may allow to execute all matching algorithms in sequential manner, or it may execute some of algorithms in defined sequence. The system provides different matching algorithms, such as string matching, label matching, child matching (means internal Childs of source and target), and property matching, these are well-known algorithms in ontological mappings. But the existing matching systems lack to inset the explicit semantics into the space vectors, therefore, we inset the explicit semantics into matching algorithms. The proposed system is evaluated to map localized concepts of a DCM of our collaborative hospital with standard terminology SNOMED CT. The localized DCM covered the concepts of head and neck cancer. The system mapped the concepts of DCM with SNOMED CT with 95% of precision, 92% of recall, and 93% F-measure. The proposed system only focuses on high precision and recall of the mapping.

4.1.1 Architectural View

The significant feature of semantic reconciliation methodology is the inclusion of semantics into the vectors of the source and target terminologies. The existing algorithms only focused on the internal semantics as child, siblings, and parents' similarity matching within the source and target terminologies. In addition to the external semantics, the proposed system insets the external semantics of the matching concepts in the form of their synonyms, hypernyms, hyponyms, and meronyms from semantically rich libraries. We used well-known libraries WordNet and ConceptsNet5 for including the external semantics. The system transmutes the short form of acronyms into their extended form using acronyms library. The DCM, developed for our collaborative hospital, contains acronyms for some concepts; acronyms highly effect the accuracy of matching algorithms. Therefore, the proposed system included the acronyms inset using the All-Acronyms library.

We designed and implemented a library to orchestrate multiple matching algorithms based on the selected algorithm execution strategy. The strategy can be selected to execute all matching algorithms sequentially or to execute some selected algorithms in a specific manner. Figure 4.3 demonstrates the reconciliation model to map the standard terminology and DCM. Following are the detail description of the reconciliation model. *Execution Control* fetches the concepts of DCM and standard terminology SNOMED CT. When concepts exist in both of the terminologies then different matching algorithms are executed to find similarity score. If the similarity score is higher than threshold value (0.8) then source and target concepts are considered as mapped concepts. In

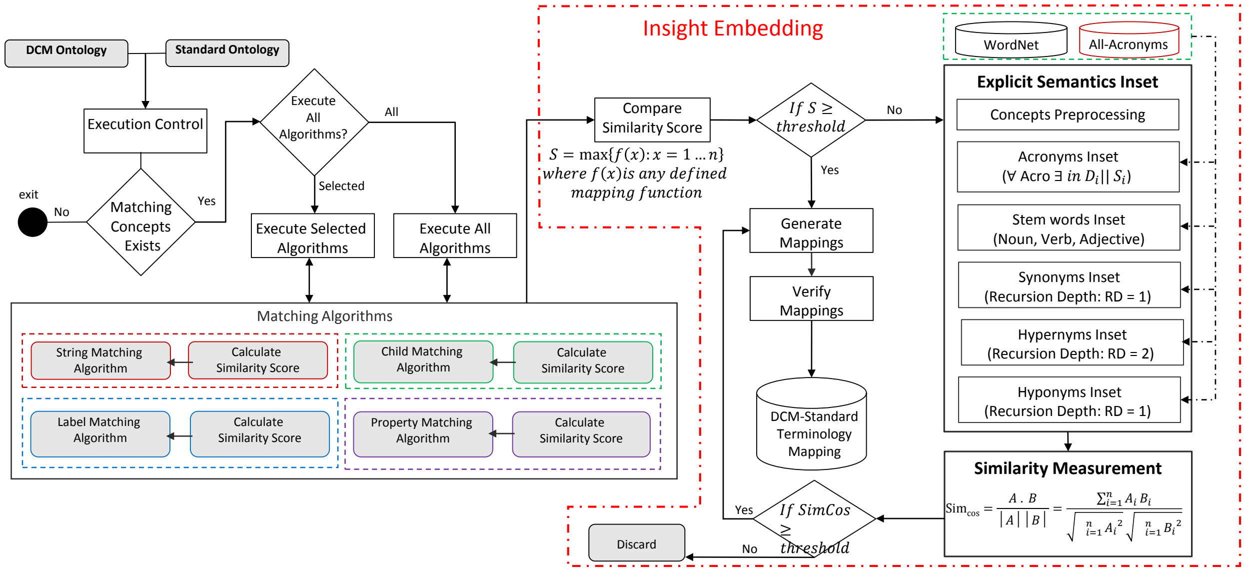

Figure 4.3: Semantic reconciliation for DCM-Standard Terminology

contrast, if similarity score is less than threshold value then we inset the explicit semantics into the space vector.

Explicit Semantic Inset for insertion the external semantics to the matching concepts' space vectors. In Explicit Semantic Inset, the concepts are preprocessed for the basic operations such as tokenize the concepts and remove the stop words, if exist; using the Concepts Preprocessing component. In second step, the *Acronym Inset* transmutes the short form of acronyms, if exist; to the corresponding extended forms using All-Acronyms library [111]. For instance, the concept "EB Virus "is transformed to "Epstein-Barr Virus". In third step, Stem words Inset transforms the concepts to their stem words. We focused on the stem transformation of nouns, verb, and adjectives because these parts of speech are mostly used in the terminology concepts.

In fourth step, we include the synonyms of the concepts using *Synonym Inset* component. We limited the synonym recursion depth up to second level of hierarchy due to performance with respect to handling the multi-dimensionality of the concepts. In fifth step, the semantics with respect to hypernyms are included into the concepts vector using *Hypernyms Inset* component. In case of hypernyms insertion, we extended the recursion depth up to two, because the multi-dimensionality cannot be effected by two hypernym concepts. In sixth step, the external semantics with respect hyponyms are inserted into the concepts by *Hyponyms Inset*. There is a chance of the existence of many Childs for the matching concepts. Therefore, the recursion depth is selected up to first level of child concepts and it regulates the multi-dimensionality to some extent. In final step of Explicit Semantics Inset, we included the meronyms of the matching concepts using *Meronyms Inset*. Meronyms may not be available for each matching concept; in unavailability case, the system discards the meronyms insertion. We selected the recursion depth as first level for the meronyms insertion.

After the insertion of explicit semantics, a strategy is built for the execution of matching algorithms. The system has two options to execute algorithms; a) execute all algorithms sequentially, b) execute only selected algorithms; from the Matching Algorithms Library. In the proposed system, we have String Matching, Label Matching, Child Matching, Parent Matching, and Property Matching algorithms. Based on the selected strategy, the system executes the algorithms. Each algorithm calculates the similarity score and provides it to the Generate Mappings component. This component evaluates and compares the similarity score with the defined threshold value "0.8", which is a recommended threshold value for concept matching in medical domain [112]. When the similarity score is greater than or equal to the threshold value then the concepts are considered as matched concepts, otherwise concepts are unmatched and system is executed for another iteration. The Verify Mappings component verifies the mapped concepts, when a single concept is mapped with multiple concepts of SNOMED CT then Verify Mappings gives alert to physicians for verification and final decision. Based on verification criteria, the mapped concepts are stored into the DCM-Standard Terminology Mapping repository otherwise the concepts are discarded.

4.1.2 Matching Algorithms

The Execution Control allows to select some particular matching algorithms or all algorithms for execution in sequential form. Firstly, the string matching algorithm provides matching concepts based on characters involved in the concepts. This includes prefix, suffix, edit distance, n-gram and others string matching algorithms. The equation 4.1, describes the function $f(x)$ performing the string matching on two concepts C_i and C_j in two different ontologies source O_s and target

O_t, respectively. Different string matching techniques are used to find similarity score value.

$$\begin{aligned} f(x) \leftarrow C_i(O_s) \bigcap C_j(O_t) \forall S_k \therefore C_i \in O_s, C_j \in O_t \\ M_{apped} C_{oncepts} = \mathbb{R}_{i=1}^{C_n} . \mathbb{R}_{j=1}^{C_n} f(x) \\ where S_k \in \{\textit{set of string matching algorithms}\}. \end{aligned} \tag{4.1}$$

In label matching, the system maps concept C_i and C_j of source O_s and target O_t ontologies based on the assigned labels. The assigned labels in ontologies may contain some special characters and stop words, therefore, it is required to remove before mapping. The function $h(x)$ in equation 4.2 removes the stem and special characters from labels and concatenate with space.

$$\begin{aligned} & h(x) \leftarrow removeSpecialAndStem(C_i(O_s).label, C_j(O_t).label) \therefore C_i \in O_s, C_j \in O_t \\ & Concatenated(h(x)) \leftarrow \sum_{i,j}^{C_n} h(x) \end{aligned} \tag{4.2}$$

After removing stem and special characters, the concepts C_i and C_j in Equation 4.1 is replaced with their labels such as $C_i.label$ and $C_j.label$, it becomes as Equation 4.3

$$\begin{aligned} f(x) \leftarrow C_i(O_s).label \bigcap C_j(O_t).label \forall S_k \therefore C_i \in O_s, C_j \in O_t \\ M_{apped} C_{oncepts} = \mathbb{R}_{i=1}^{C_n} . \mathbb{R}_{j=1}^{C_n} f(x) \\ where S_k \in \{\textit{set of string matching algorithms}\}. \end{aligned} \tag{4.3}$$

In child matching algorithm, we map the child concepts of source C_i and C_j target concepts. Internally, the child mapping use the string matching algorithm of Equation 4.1 to find the same childs of the input concepts. Algorithm 1 describes the child matching process with the help of Equation 4.4. If the number of mapped child is greater than threshold value then the input concepts C_i and C_j are considered as matched concepts.

$$\begin{aligned} f(x) \leftarrow Child_{i.c}(O_s) \bigcap Child_{j.c}(O_t) \forall S_k \therefore Child_{i.c} \in O_s, Child_{j.c} \in O_t \\ M_{apped} C_{oncepts} = \mathbb{R}_{i=1}^{C_n} . \mathbb{R}_{j=1}^{C_n} f(x) \\ where S_k \in \{\textit{set of string matching algorithms}\}. \end{aligned} \tag{4.4}$$

Algorithm 1: child matching algorithm

Input : C_i ***of*** O_s Source Ontology Concept
C_j ***of*** O_t Target Ontology
Result: $M_{apped}C_{oncepts}$

Initialization: NumMappedChild = 0;
foreach $Child_{i.c}$ *of* C_i *in* O_s **do**
  foreach $Child_{j.c}$ *of* C_j *in* O_t **do**
    if $Child_{i.c}$ *is equal to* $Child_{j.c}$ **then**
      $f(x) \leftarrow Child_{i.c}(O_s) \bigcap Child_{j.c}(O_t) \forall S_k \therefore Child_{i.c} \in O_s, Child_{j.c} \in O_t$;
      NumMappedChild = NumMappedChild + 1;
      Break;
    end
  end
  if $NumMappedChild$ *is greater than or equal to* $ThresholdValue$ **then**
    $M_{apped}C_{oncepts} \leftarrow C_i(O_s) \bigcap C_j(O_t)$;
    Break;
  end
end

In ontology matching, properties has very important role in describing a particular concept. If some properties of source C_i and target C_j concepts, and the number of mapped properties are greater or equal to threshold value then those concepts are considered as mapped concepts. The process of Algorithm 1 can utilize to map the properties mapping, but the only difference is to check the properties instead of child. The Equation 4.5 shows the core of property matching of source C_i and C_j target concepts.

$$f(x) \leftarrow Property_{i.c}(O_s) \bigcap Property_{j.c}(O_t) \forall S_k \therefore Property_{i.c} \in O_s, Property_{j.c} \in O_t$$
$$M_{apped}C_{oncepts} = \mathbb{R}_{i=1}^{C_n}.\mathbb{R}_{j=1}^{C_n} f(x) \quad (4.5)$$
$$where S_k \in \{\textit{set of string matching algorithms}\}.$$

After finding similarity score using the aforementioned corresponding equations and algorithms, the similarity score is compared with threshold value, if it is less than threshold value then our proposed technique embed the explicit semantics into a space vector with acronyms, synonyms, hypernyms, and hyponyms. This process enrich the space vector of the source C_i and target C_j concepts with their explicit semantics. The enriched space vector enhance the similarity score of the mapping concepts. The semantically enriched space vectors for source C_i and target C_j concepts are depicted in Equation 4.6 and Equation 4.7, respectively.

$$SV(C_i) = [Stem(C_i), Acro(C_i), \sum_{k=0}^{n} Syn(C_i), \sum_{k=0}^{n} Hyper(C_i), \sum_{k=0}^{n} Hypo(C_i), \sum_{k=0}^{n} Mero(C_i)] \tag{4.6}$$

$$SV(C_j) = [Stem(C_j), Acro(C_j), \sum_{k=0}^{n} Syn(C_j), \sum_{k=0}^{n} Hyper(C_j), \sum_{k=0}^{n} Hypo(C_j), \sum_{k=0}^{n} Mero(C_j)] \tag{4.7}$$

Where $\sum_{k=0}^{n}$ is the collection of dynamic number of corresponding concepts such as number of synonyms, number of hypernyms, number of hyponyms, and number of meronyms of source C_i and target C_j concepts. The space vectors $SV(C_i)$ and $SV(C_j)$ of corresponding source C_i and target C_j concepts are used to find the final similarity score, which is explained in the following section. Algorithm 2 shows the complete process of DCM and standard terminology concepts.

4.1.3 Scenario: Finding Final Similarity Score

The proposed system insets the external semantics before the execution of matching algorithms and finding the similarity score. This approach enhanced the mapping accuracy and resolved the issue of multiple matched concepts. We extended the standard cosine similarity formula (Equation 4.8), based on the semantics insertion.

$$CosineSimilarity = \frac{A.B}{\|A\|.\|B\|} = \frac{\sum_{i=1}^{n} A_i.B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \tag{4.8}$$

Algorithm 2: DCM and Standard Terminology mapping algorithm

Input : O_s Source Ontology
O_t Target Ontology
Result: $M_{apped}C_{oncepts}$

Initialization: InitialSimilarityScore = 0.0;
Initialization: FinalSimilarityScore = 0.0;
foreach C_i *of* O_s **do**
  foreach C_j *of* O_t **do**
    $stringSimilarity$ = stringMatching(C_i, C_j);
    $childSimilarity$ = childgMatching(C_i, C_j);
    $labelSimilarity$ = labelMatching(C_i, C_j);
    $propertySimilarity$ = propertyMatching(C_i, C_j);
    InitialSimilarityScore = $max(stringSimilarity, childSimilarity, labelSimilarity, propertySimilarity)$;
    if *InitialSimilarityScore is less than* $ThresholdValue$ **then**
      $SV(C_i)$ = preprocessConcepts(C_i);
      $SV(C_i)$ = $SV(C_i)$ + InsertExplicitSemantics(C_i);
      $SV(C_j)$ = preprocessConcepts(C_j);
      $SV(C_j)$ = $SV(C_j)$ + InsertExplicitSemantics(C_j);
      $Cosine_{sim} = \frac{\sum_{i=1}^{n} SV(C_i).SV(C_j)}{\sqrt{\sum_{i=1}^{n} SV(C_i)^2}\sqrt{\sum_{i=1}^{n} SV(C_j)^2}}$;
      FinalSimilarityScore = $Cosine_{sim}$;
      if *FinalSimilarityScore is greater than or equal to* $ThresholdValue$ **then**
        $M_{apped}C_{oncepts} \leftarrow C_i(O_s) \bigcap C_j(O_t)$;
        Break;
      end
    end
    else
      $M_{apped}C_{oncepts} \leftarrow C_i(O_s) \bigcap C_j(O_t)$;
      Break;
    end
  end
end

We added the union of stems, synonyms, hypernyms, hyponyms, and meronyms into Equation 4.8 and extended to equation in Figure 4.4. As an example, one DCM concept “Smoking status” is mapped with three different SNOMED CT concepts “Smoking status at 4 week”, “Smoking status at 52 week”, and “Smoking monitoring status” according to the standard cosine similarity in Equation 1 with similarity score 0.816. The same similarity score for three concepts creates misperception in matching the exact concept. The proposed system resolved this misperception with the insertion of explicit semantics using equation in Figure 4.4. Our approach calculated the similarity score 0.739 for “Smoking status at 4 week” and “Smoking status at 52 week”, while 0.926 is calculated for “Smoking monitoring status”. Therefore, it is easily distinguishable and it is considered as matched concepts. The inserted semantics are shown Figure 4.4.

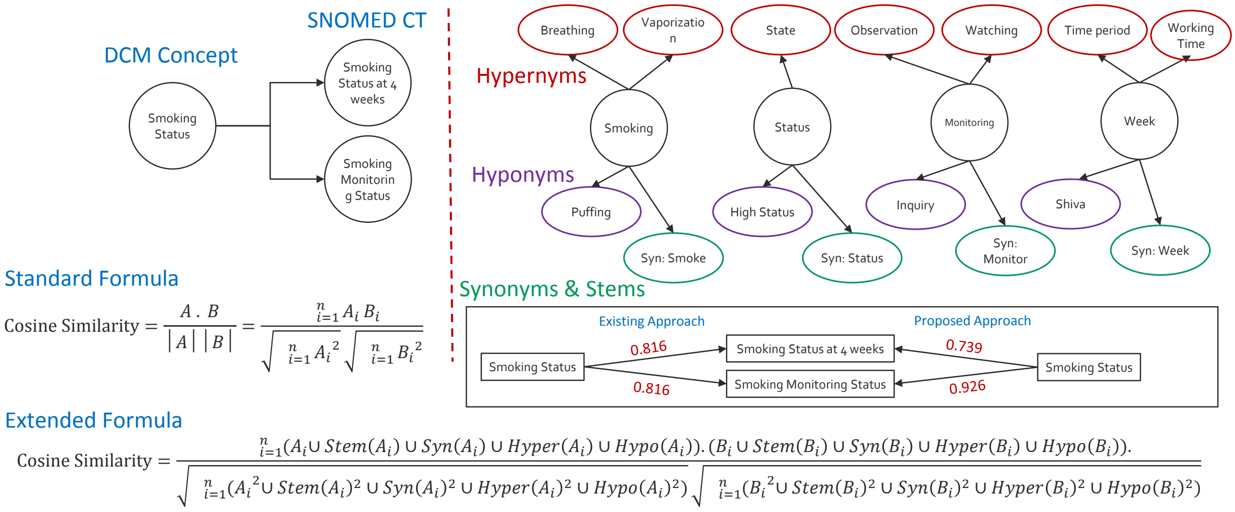

Figure 4.4: Example of external semantic insertion

4.2 Standard Terminology and Data Model Mapping

In the start of this chapter, a production rule is transformed into standard format with emergence of standard data model vMR schema classes and standard terminology SNOMED CT codes, as depicted in Figure 4.2. The concepts used in production rules such as *disease, head & neck cancer, age, treatment plan, and surgery*, are localized concepts of DCM. The left side concept of each fact in the rule transformed into corresponding standard data model vMR schema classes with attributes such as *ObservationResult.observationFocus*. While the right side concept of each fact transformed into standard terminology code such as *64572001*. Therefore, the proposed system requires mapping among standard terminology concepts and standard data model classes with attributes.

The standard data model vMR contains approximately 49 classes with hundreds of attributes, on the other hand, SNOMED CT contains more than 0.3 million concepts with 19 top hierarchical concepts. A single attribute of vMR class can be bind with some particular concepts in a single or multiple hierarchies of the SNOMED CT. It is difficult to map vMR classes and SNOMED CT hierarchical concepts using the aforementioned matching algorithms in previous section 4.1. The standard concepts can be mapped based on the concepts definitions published in the corresponding specification documents such as SNOMED CT specification document *SNOMED CT Starter Guide* [108] and vMR specification document *HL7 Virtual Medical Record for Clinical Decision Support (vMR-CDS) Logical Model, Release 2* [107]. Therefore, we introduced definition-based mapping in the ontology matching algorithms, which maps the standard concepts based on their definitions. The definition-based mapping is different than the semantic mapping. In definition-based mapping, we mapped the concepts definitions, which are defined in specification document, while the semantic mapping focuses on the internal semantics of concepts within the particular ontology. Figure 4.5 illustrates the abstract view of the reasons of including definition-based mapping in the existing ontology matching algorithms.

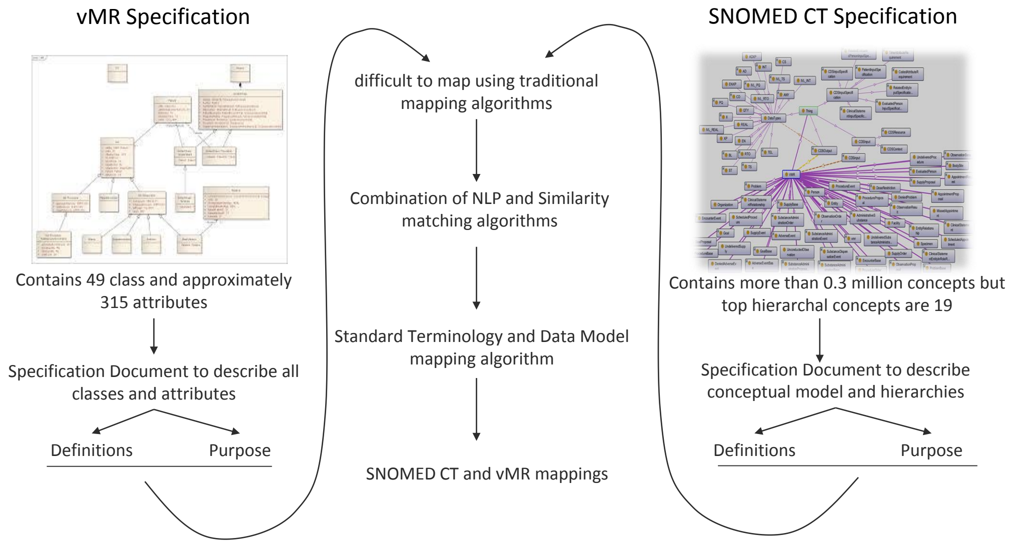

Figure 4.5: Abstract view of standard terminology and data model mapping

In proposed technique, we map the definition of vMR classes and their attributes with definition of SNOMED CT top hierarchical concepts. The cardinality of mapping is many to many, a single vMR class and attribute can map with multiple top hierarchical concepts of SNOMED CT. Similarly, a single top hierarchical concept of SNOMED CT can be mapped with multiple vMR classes and attributes.

4.2.1 Architectural View

In previous section 4.1, we inset the explicit semantics into space vectors of the source concept and target concept to semantically enrich the corresponding vectors for enhancing the matching algorithms' recall and precision. The similar approach of explicit semantics embedding is also used in standard terminology and data model mappings. Additionally, we inset the implicit semantics into the vectors after preprocessing the definitions of vMR classes and top hierarchical concepts. The explicit semantics is defined in definition 4.2.1, while implicit semantics is defined in definition 4.2.2.

Definition 4.2.1: Explicit Semantics: The explicit semantics of a concept C_i in ontology O_s is $ES(C_i)$, which is the collection of synonyms ($C_i \in Syn(C_i)$), its generalization ($C_i \in Hyper(C_i)$), and specialization ($C_i \in Hypo(C_i)$) of concept.

Definition 4.2.2: Implicit Semantics: The implicit semantics of a concept C_i in ontology O_s is $IS(C_i)$, which is the collection of its lexical chain $lc(C_i)$ as the sum of all implicit relationships $lc(C_i) = \sum RelatedTerms(C_i) + \sum TypeOf(C_i) + \sum PartOf(C_i) + \sum SeeAlso(C_i)$.

We proposed the definition-based algorithm to integrate with systems to find the similarity of concepts based on definitions in the corresponding specification documents. Figure 4.6 depicts the semantic reconciliation process of standard terminology and standard data model mapping. We extract the definition of each concept from standard terminology and data model from specification document using *Extract Definitions*, which are already identified in documents. After definition extraction, we prepare the space vector (SV) for each of the concepts using *Create Word Vector*. The created vectors are use to preprocess with concept extraction, removing stop words, and finding stem words in the sub process of *Explicit Semantics Inset*. The process of explicit semantics

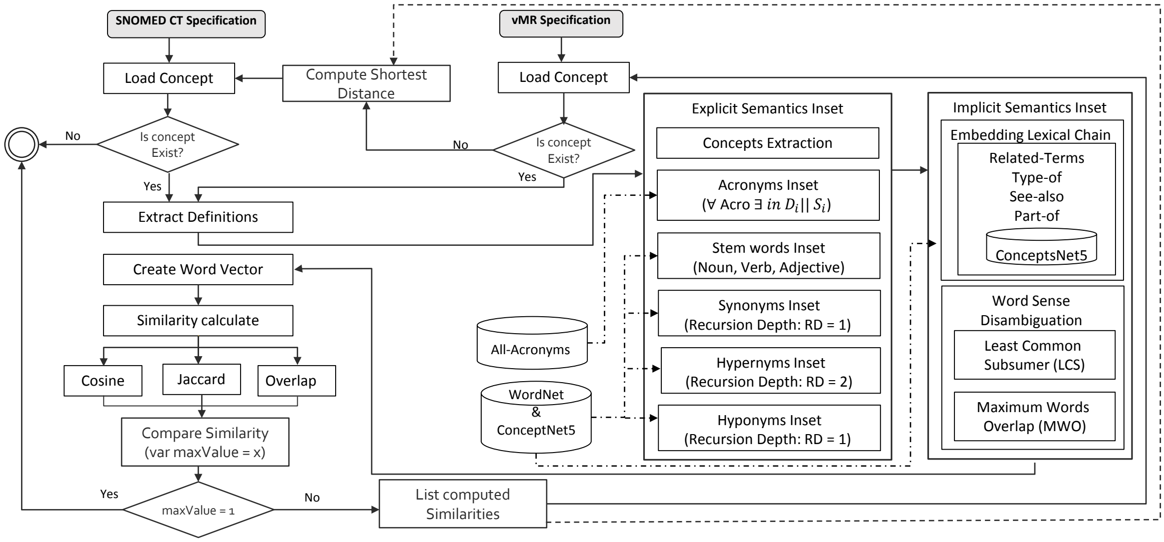

Figure 4.6: Semantic reconciliation for standard terminology and data model

embedding is similar that already described in previous section 4.1. Additionally, we inset the implicit semantics into space vector (SV) of each concept using *Implicit Semantics Inset*.

As mentioned in definition 4.2.1, the explicit semantics are directly related to the concepts synonyms, its generalized concepts, and specialized concepts. However, according to definition 4.2.2, we embed the lexical chain with implicit relationship of a particular concept with other concepts such as *Related-Terms*, *Type-of*, *Part-of*, and *See-also*. We used a well-known repository of concepts network, called *ConceptsNet*. The amalgamation of different specialized, generalized, and implicit relationship concepts lead to word sense disambiguation problem [113]. Currently, there are some existing solutions are available for the word sense disambiguation problem. However, we used two solutions *Maximum Words Overlap (MWO)* [114] and *Least Common Subsumer (LCS)* [115], due to their better results in word disambiguation algorithms.

The space vector with explicit semantics and explicit semantic are send to *Similarity Calculate* module for finding the similarity between vectors. This module calculates similarities with three different similarity finding algorithms *Cosine Similarity*, *Jaccard Similarity*, and *Overlap Similarity*. We selected these three similarity algorithms due to their high performance in precision. Sometimes, all three algorithm give very same similarity score, but usually the similarity score are different. In case of different similarity score values, we consider the maximum score value as a final similarity score. The *Compare Similarity* module is responsible for finding maximum value of similarity score. We add that score values with corresponding concepts into list of computed similarities as *List Computed Similarities*. After complete all iterations for a single concept, we choose the top 2 concepts as matched concepts and store those concepts as one-to-many mappings list in the repository. The same process will be continue till the end of all concepts of standard data model repository.

4.2.2 Algorithmic View

As we discussed in previous section, that ontology matching techniques usually lack definition-based matching, however, our proposed algorithm takes two standard ontologies source ontology O_s and target ontology O_t with specification and definition of concepts. Each set of ontology $O_s = \{C_1, C_2, C_3, .., C_i, ..., C_n\}$, and $O_t = \{C_1, C_2, C_3, .., C_j ..., C_n\}$ contains identified definitions of the concepts. The architecture in Figure 4.6 is algorithmically depicted in Algorithm 3.

After extracting definition of each concepts, the preprocessed definition is stored into a space vectors $SV(C_i)$ and $SV(C_j)$ for source and target concepts. In processing, we remove the stop words and special characters, and transmute the short form of different abbreviations, if exists. We insert the explicit semantics into space vectors with same equations 4.6 and 4.7 for space vectors $ESV(C_i)$ and $ESV(C_j)$, respectively, as described in previous section 4.1.

The $InsertImplicitSemantics(C_j)$ function is inset the lexical chain of implicit relationships with different properties such as *Related-Terms*, *Type-of*, *Part-of*, and *See-also*. The two implicit space vector $ISV(C_i)$ for source and $ISV(C_j)$ for target concepts are prepared as in Equation 4.9 and Equation 4.10, respectively. These relationships are fetch from online repository ConceptsNet5 for each corresponding concept. The implicit relationships decrease the computational performance but it increase the precision and recall of the matching algorithms, which is our required criteria to find more precise concepts. When the space vector preparation is completed then we find similarity scores using different similarity algorithms.

$$ISV(C_i) = lc(\sum_{k=0}^{n} RelatedTerms(C_i), \sum_{k=0}^{n} TypeOf(C_i), \sum_{k=0}^{n} PartOf(C_i), \sum_{k=0}^{n} SeeAlso(C_i))$$

Algorithm 3: Standard Terminology and data model mapping algorithm

Input : O_s Source Ontology as data model specification
O_t Target Ontology as standard terminology specification
$O_s = \{C_1, C_2, C_3, .., C_i, ..., C_n\}$
$O_t = \{\}C_1, C_2, C_3, .., C_j..., C_n\}$
Result: $M_{apped}C_{oncepts}$

Initialization: FinalSimilarityScore = 0.0;
foreach C_i *of* O_s **do**
    Definition(C_i) = ExtractDefinition(C_i);
    $SV(C_i)$ = preprocessDefinition(C_i);
    $SV(C_i)$ = $SV(C_i)$ + InsertExplicitSemantics(C_i);
    $SV(C_i)$ = $SV(C_i)$ + InsertImplicitSemantics(C_i);
    foreach C_j *of* O_t **do**
        Definition(C_j) = ExtractDefinition(C_j);
        $SV(C_j)$ = preprocessDefinition(C_j);
        $SV(C_j)$ = $SV(C_j)$ + InsertExplicitSemantics(C_j);
        $SV(C_j)$ = $SV(C_j)$ + InsertImplicitSemantics(C_j);
    end
    $Cosine_{sim} = \frac{\sum_{i=1}^{n} SV(C_i).SV(C_j)}{\sqrt{\sum_{i=1}^{n} SV(C_i)^2}\sqrt{\sum_{i=1}^{n} SV(C_j)^2}}$;
    $Jaccard_{sim} = \frac{|SV(C_i) \cap SV(C_j)|}{|SV(C_i) \cup SV(C_j)|}$;
    $Overlap_{sim} = \frac{|SV(C_i) \cap SV(C_j)|}{min(|SV(C_i)|, |SV(C_j)|)}$;
    FinalSimilarityScore = $max(Cosine_{sim}, Jaccard_{sim}, Overlap_{sim})$;
    if *FinalSimilarityScore* $isgreaterthanorequaltoThresholdValue$ **then**
        $M_{apped}C_{oncepts} \leftarrow C_i(O_s) \bigcap C_j(O_t)$;
        Break;
    end
end

(4.9)

$$ISV(C_j) = lc(\sum_{k=0}^{n} RelatedTerms(C_j), \sum_{k=0}^{n} TypeOf(C_j), \sum_{k=0}^{n} PartOf(C_j), \sum_{k=0}^{n} SeeAlso(C_j)) \tag{4.10}$$

Once we get four vectors, source explicit space vector $ESV(C_i)$ and implicit space vector $ISV(C_i)$, and target explicit space vector $ESV(C_j)$ and implicit space vector $ISV(C_j)$ then we integrate these vectors to make it appropriate format for similarity matching, using equations 4.11 and 4.12.

$$SV(C_i) = ESV(C_i) + ISV(C_i) \tag{4.11}$$

$$SV(C_j) = ESV(C_i) + ISV(C_j) \tag{4.12}$$

We use three similarity algorithms Cosine Similarity ($Cosine_{sim}$), Jaccard Similarity ($Jaccard_{sim}$), and Overlap Similarity ($Overlap_{sim}$) using equations 4.13, 4.14, and 4.15, respectively.

$$Cosine_{sim} = \frac{\sum_{i=1}^{n} SV(C_i).SV(C_j)}{\sqrt{\sum_{i=1}^{n} SV(C_i)^2}\sqrt{\sum_{i=1}^{n} SV(C_j)^2}} \tag{4.13}$$

$$Jaccard_{sim} = \frac{|SV(C_i) \cap SV(C_j)|}{|SV(C_i) \cup SV(C_j)|} \tag{4.14}$$

$$Overlap_{sim} = \frac{|SV(C_i) \cap SV(C_j)|}{min(|SV(C_i)|, |SV(C_j)|)} \tag{4.15}$$

Finally, we consider the maximum similarity score as a final value, and that value is compared with threshold value. Final similarity score is identified using Equation 4.16. If final similarity value is greater than or equal to threshold value then the concepts are considered as matched concepts otherwise as unmatched concepts.

$$FinalSimilarityScore = max(Cosine_{sim}, Jaccard_{sim}, Overlap_{sim}) \tag{4.16}$$

4.3 DCM and Standard Data Model Mapping

The third type of mappings provided by SRM are mappings among localized concepts of DCM and standard data model such vMR. Once we get mappings among DCM and standard terminology and mappings among standard terminology and data model, then we can easily get the mapping among DCM and standard data model using transitivity law in Equation 3.1 in chapter 3. Figure 4.7 shows the DCM and standard data model mapping using transitivity law.

The resultant mapping are shown in Figure 4.8, which illustrates an example of DCM-SNOMED, vMR-SNOMED, and DCM-vMR partial mappings for three DCM concepts (i.e., *surgical history*, *potentially malignant disorders*, and *psychiatric disease history*) that belong to the *Subjective* part of DCM with some partial mappings shown in Table 4.1. In Table 4.1, columns 1, 2, and 3 show some of the mappings between DCM, SNOMED CT, and vMR concepts, while columns 4, 5, and 6 (*DCM Values Set*, *SNOMED CT Code (for Values set)*, and *vMR Concept Attributes (for value set)*) show the mapping of corresponding values sets of DCM, SNOMED CT, and vMR concepts, respectively. Table 4.1 partially depicts the three types of mappings in SRM.

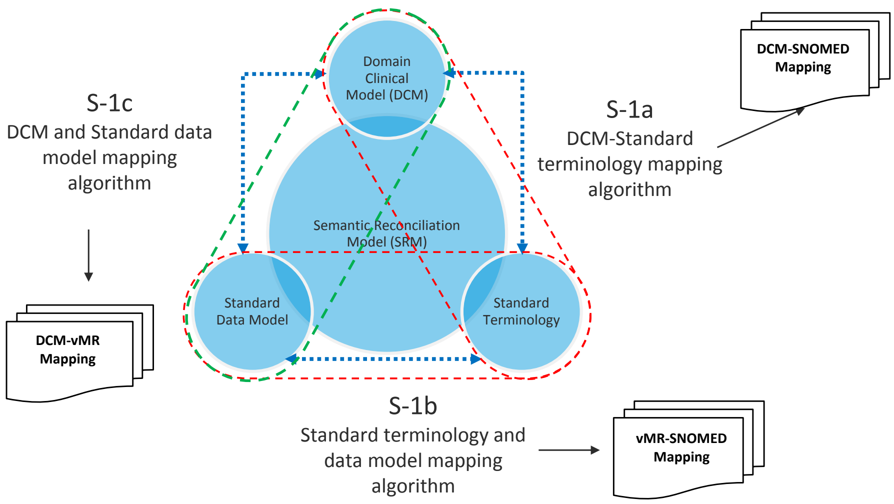

Figure 4.7: DCM and standard data model mapping using transitivity law.

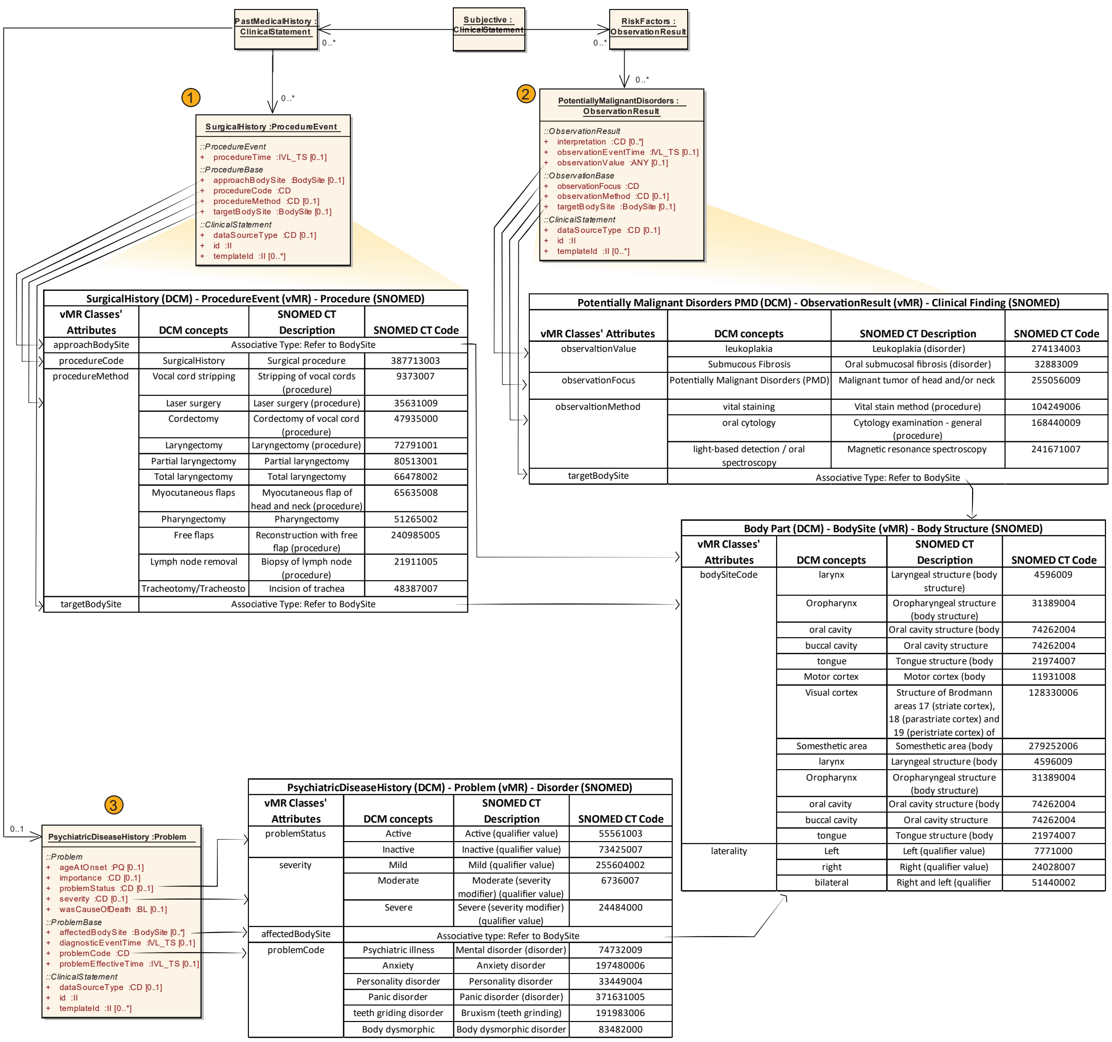

Figure 4.8: SRM Example: Attribute-level mappings of vMR, DCM, and SNOMED CT.

Table 4.1: **DCM concepts mapping to corresponding vMR and SNOMED CT concepts.**

DCM Concept	SNOMED CT Concept	vMR Concepts	DCM Values Set	SNOMED CT Code (for Values Set)	vMR Concepts Attributes (for Values Set)
Clinical Stage T	385356007 Tumor stage finding (finding)	ObservationResult (observationFocus)	T0	58790005	observationValue
			T1	23351008	
			T2	67673008	
			T3	14410001	
			T4/T4a	65565005	
			T4b	396731008	
			Tx	43189003	
Clinical Stage N	385382003 Node (category finding (finding), N stage finding, Node category finding, Node stage finding)	ObservationResult (observationFocus)	N0	62455006	observationValue
			N1	53623008	
			N2	46059003	
			N3	5856006	
			N2a	261967001	
			N2b	370008004	
			N2b	370010002	
			Nx	79420006	
Clinical Stage S	80631005 Clinical stage finding (finding), Clinical stage finding	ObservationResult (observationFocus)	I	13104003	observationValue
			II	60333009	
			III	50283003	
			IV/IV A	2640006	
			IV B	1523005	
			IV C	33177002	
Treatment Intent	395077000 Treatment intent (situation)	ProcedureEvent (procedureCode)	Radical	27762005	procedureMethod
			Palliative	363676003	
			Consultation	11429006	
Histology Description	250537006 Histopathology finding	Problem (problemCode)	Squamous cell carcinoma	402815007	problemCode
			Small cell carcinoma	74364000	
			Carcinoma NOS	68453008	
			Adenocarcinoma	35917007	
			Adenoid cystic carcinoma	11671000	
			Adenoid cystic carcinoma	1338007	
			Squamous cell carcinoma in situ	59529006	
			Verrucous carcinoma	89906000	
			Malignant melanoma	2092003	
			Pleomorphic adenoma	8360001	
			Spindle cell carcinoma	65692009	
			Ameloblastoma, malignant	88253001	
			Adenoid squamous cell carcinoma	85956000	
			nasopharyngeal carcinoma	449248000	

Continued on next page

Table 4.1 – *Continued from previous page*

DCM Concept	SNOMED CT Concept	vMR Concepts	DCM Values Set	SNOMED CT Code (for Values set)	vMR Concepts attributes (for Values set)
			Sebaceous adenocarcinoma	54734006	
			Sarcoma, not otherwise specified	397355008	
			Plasmacytoma, not otherwise specified	415112005	
			Mucoepidermoid carcinoma	4079000	
Treatment Plan	1. 413737006 Cancer hospital treatment completed (situation) 2. 225292002 Developing a treatment plan (procedure)	ProcedureEvent (procedureCode)	Chemotherapy	367336001	procedureMethod
			CRT (Chemoradiotherapy)	703423002	
			RT (Radiotherapy)	108290001	
			Surgery	387713003	
			Induction Chemotherapy	450827009	

4.4 Summary

In this chapter, we discussed the *Schema-Data Level Semantic Reconciliation* to give mapping files as a prerequisites for creating shareable and interoperable knowledge. It generates three different types of mapping files such as DCM-Standard Terminology mappings, Standard Terminology-Standard Data Model mappings, and DCM-Standard data model mappings.

Chapter 5

SRM: Structure Level Semantic Reconciliation

The clinical domain has many standards to represent and share data, information, and knowledge. The communication standards are also very helpful to disseminate clinical knowledge. Therefore, knowledge shareablity can be achieved through a standard knowledge representation. The well-known clinical knowledge representations are Arden Syntax Medical Logic Module (MLM) [20], Guideline Interchange Format (GLIF) [70], Shareable Active Guideline Environment (SAGE) [71]. Our proposed methodology has flexible slots generation technique to generate input data, logic, and action, which are common parts in each knowledge representation. In this thesis, we target the Arden Syntax Medical Logic Module (MLM), which is a standard unit of clinical knowledge [20]. The Health Level-7 (HL7) community has designed Arden Syntax-based MLM to share and disseminate the clinical knowledge among different medical institutions. HL7 Arden Syntax is an ANSI standard to provide a comprehensive structure and syntax for representing clinical knowledge [21, 22]. We explained the Structure Level Semantic Reconciliation in chapter 3, which automatically creates the shareable knowledge with high level of abstraction.

5.1 Standard Knowledge Representation

We used HL7 Arden Syntax MLM to represent the shareable knowledge to disseminate among heterogeneous clinical institution. The standard MLM has a specific syntax of language, called Arden Syntax, and it has a specific structure to represent a knowledge. Figure 5.1 shows a standard format of structure and syntax of MLM. Standard MLM has three main categories; maintenance, library and knowledge. These are used to represent medical knowledge in the form of Arden Syntax artifacts.

Structure and Syntax

```
maintenance:
        title: Palliative Treatment By Physician 3;;
        mlmname: Palliative Treatment By Physician 3;;
        arden: Arden Syntax V2.7;;
        version: Version 2.7;;
        institution: SKMCH;;
        author: Dr. Physician 3;;
        specialist: Dr. Physician 3;;
        date: 13/01/2015;;
        validation: testing;;
Library:
 purpose:  Experimental testing;;
 explanation: Experimental testing;;
 keywords: Oral Cavity;;
 citations:   ;;
Knowledge:
         type: data driven; ;
         data:
        LET varTreatmentIntent = BE Read { Select TreatmentInten from ClientDB }
        ;;
evoke: null_event; ;
          logic:
                 if ( varTreatmentIntent is equal to Palliative)
                 {
                   Conclude true;
                 };;
          action:
                      WRITE "The recommended treatment plan is Radiotherapy"
                  at stdout_dest;      ; ;
    end;
```

Figure 5.1: Standard structure and syntax of medical logic module (MLM)

- **Maintenance**: The maintenance slot of MLM describes the meta-information about the knowledge rule and its author. For instance, it describes the title, MLM name, version of Arden Syntax, institution, author name, specialist name, date of knowledge rule creation, and validation scope.
- **Library**: The main objective of library part is to describe the purpose of the knowledge rule, and its explanation. Additionally, it also focus on the knowledge keywords and citation of the related article to the knowledge.

- **Knowledge**: The is the main slot of MLM to represent the actual parts of a standard rule. The conditions and actions of rules are stored in this knowledge slot. Internally, it has sub-categories *type*, *data*, *evoke*, *logic*, and *action*. The *data* slot represents the input data required to execute that particular rule and output data in form of recommendation, alert, guideline, and any type of decision. The *logic* is the main slot of rule, which represents the condition part of the rule. Similarly, the *action* slot describes and represents the final action of the knowledge rule.

Mostly physicians interact with the critical part of knowledge slot of MLM. The objective of HL7 Arden Syntax standard is to enable physicians to share the clinical knowledge into MLMs which are easily understandable by physicians and also executable by computer system. Despite the fact of HL7 Arden Syntax as friendly representation for physicians, still most of them feel uncomfortable to represent their knowledge. Because the syntax and structure of standard MLM is very complex, which overburden the physicians to remember syntax and structure of MLM. This barrier blocks the knowledge acquisition and adaption of healthcare standards in medical institutes and thus results in increased medical costs.

The standard structure and syntax of MLM is represented in Figure 5.1, but still this MLM is not interoperable, because localized specific concepts have used in MLM instead of standard concepts. The knowledge slot is needed to amalgamate with standard data models and terminologies for making this knowledge interoperable. For instance, a production rule ***If Treatment Intent = Palliative Then Treatment Plan = Radiotherapy*** have concepts *Treatment Intent*, *Treatment Plan*, *Palliative*, and *Radiotherapy* as schema and instance concepts. Our proposed methodology transformed the schema concepts into corresponding vMR classes and attributes, while the instance concepts replaced with standard terminology SNOMED CT codes, Figure 5.2 shows the standard format of shareable and interoperable MLM with standard concepts. The *Radiotherapy* is an output parameter, and presented as *ProcedureEvent.procedureCode* in vMR data model, while its SNOMED CT code is *108290001*. Specifically, the knowledge slot of MLM is represented in standard format.

The standard data model and terminology codes amalgamation into standard knowledge representation increases the knowledge authoring complexity. Mostly, the physicians avoid such

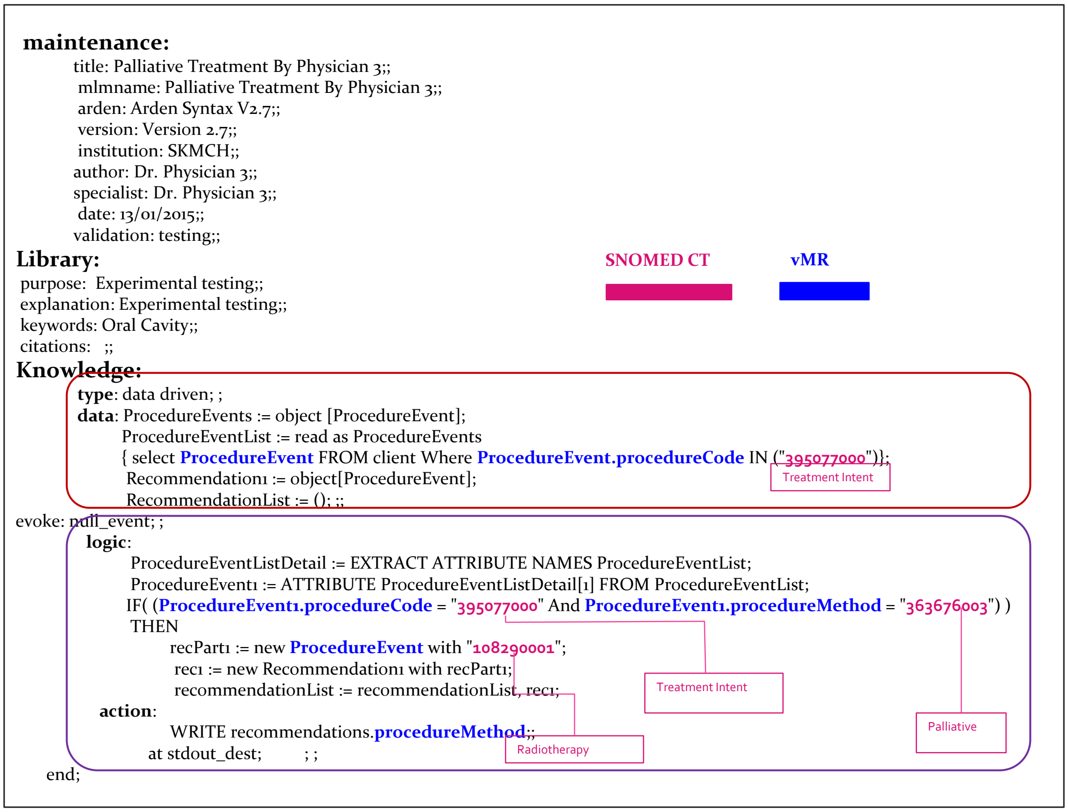

Figure 5.2: Standard structure and syntax of shareable and interoperable medical logic module (MLM)

type of overburdened activity to create knowledge from their experiences and practices. Because the standard terminology codes, standard data model classes, their attributes, syntax and structure of standard knowledge representation are difficult to remember and are considered as tedious task. Therefore, high level abstraction is needed to hide this structural, syntax, code, and classes/attributes complexity from physicians. Our proposed SRM is equipped with Structure Level Semantic Reconciliation process, which hides this complexity, and provides a very high level abstraction to physicians for creating knowledge with user-friendly interfaces. The physicians are only used understandable and localized concepts of DCM to create rule with the help of Intellisense window to list down all possible values set, as shown in Figure A.3, Appendix A.1.

5.2 Automatic Generation of Shareable Knowledge

5.2.1 Architecture

We designed and developed a knowledge creation environment that transforms the production rules into shareable and interoperable knowledge representation, which is Arden MLM representation. All components in Figure 5.3, perform the transformation of knowledge into standard format. The physicians are facilitated with *Rule Editor* to write plain rule and the system transforms into shareable and interoperable MLM without the physicians' intervention with complex structure and syntax of MLM. This abstraction also protects the physicians to memorize all the SNOMED CT concepts and schema of vMR data model with its attributes.

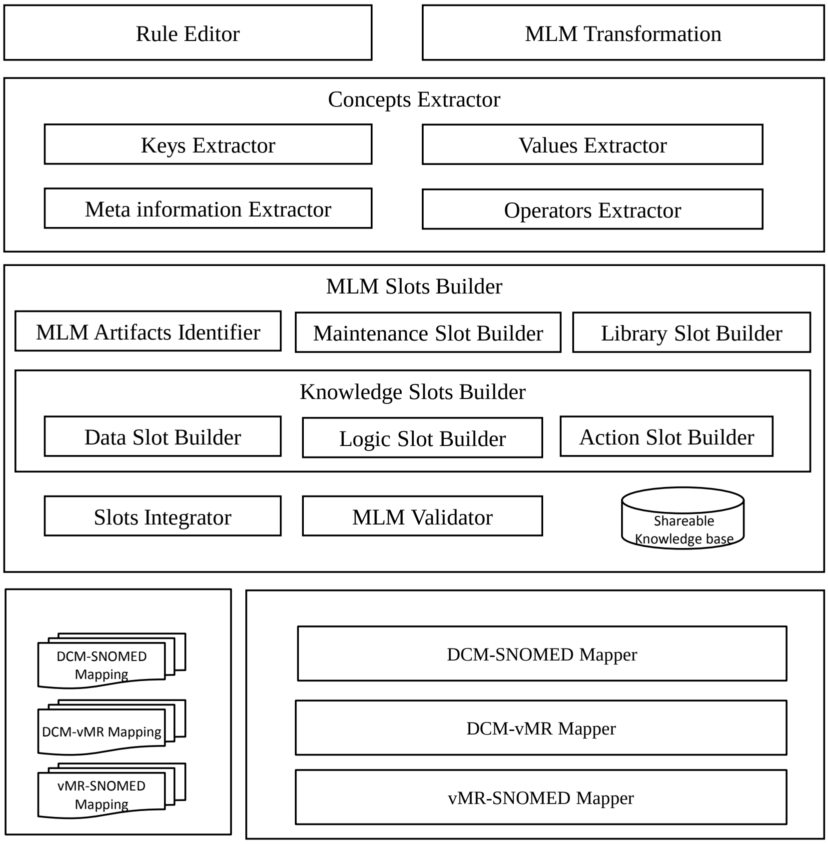

Figure 5.3: Architecture of automatic generation of shareable knowledge

The proposed system displays the DCM concepts to physicians for plain rule creation, because physicians are very familiar with local DCM concepts. While the created rules are transformed into shareable and interoperable knowledge base with amalgamation of SNOMED CT codes and vMR schema and attributes. The clinical knowledge created by this innovative approach, can be easily integrated into clinical workflows and can be shared among clinical communities of diverse cultures and regions.

Rule Editor provides a controlled vocabulary from DCM using immediate Intelli-sense window. The physicians can easily select the desired concepts as keys and values along with logical and arithmetical operators used in rules. This controlled and easy environment enhances the physicians' performance with respect to rule creation time, and it decreases the chance of errors in rules. *MLM Transformation* is the main component to perform the transformation of plain rules into Arden Syntax MLM. This component orchestrates the whole process of MLM generation. *Concepts Extractor* is responsible to extracts main artifacts of the created rules. The main ingredients of the plain rules are keys, values, and operators, and some Meta information are also associated with rules for its identification.

The standard MLM contains three main categories or slots such as Maintenance, Library, and Knowledge. The *MLM Slots Builder* is responsible to handle the aforementioned slots of MLM. The *MLM Artifacts Identifier* identifies the operators, reserved words, keywords, and syntax of MLM which are required for rule to be created. The *Slots Integrator* finally merge all the slots into a single MLM. The created or modified MLM is validated through *MLM Validator* for its standard structure, syntax and semantics of MLM. The validated MLM stored into *Shareable Knowledge Base*.

5.2.2 Algorithmic Flow

Automatic generation of MLM hides the complexity of structure and syntax of MLM, therefore, the physicians only write the rule in localized concepts as plain rule. The plain rule and three mapping files generated as output of Schema-Data Level Semantic Reconciliation Model, as mentioned in previous chapter 4. Figurer 5.4 shows the algorithmic flow of shareable and interoperable knowledge creation.

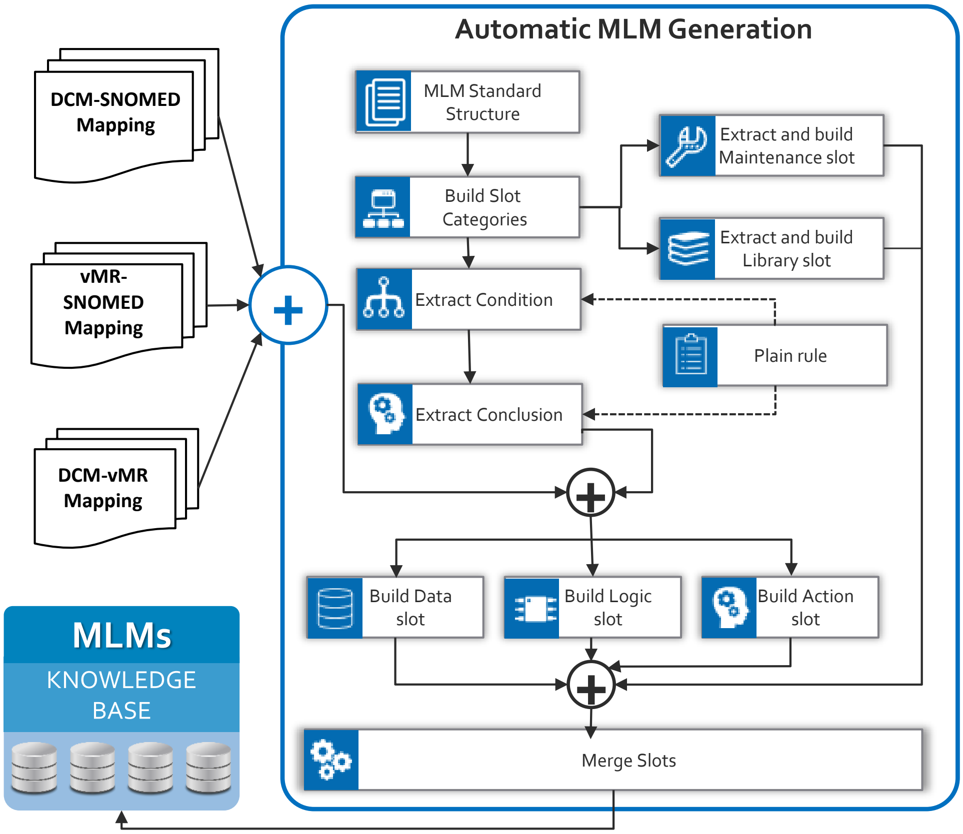

Figure 5.4: Algorithmic flow of automatic generation of shareable knowledge

- **Step 1**: The production rule is input to the MLM generation process, and mapping files DCM-SNOMED Mapping, vMR-SNOMED Mapping, and DCM-vMR Mapping are the supportive input to MLM generation.
- **Step 2**: Load the standard structure of shareable knowledge representation, which is MLM in this scenario.
- **Step 3**: Identify the required slots of knowledge and build different slot categories.
- **Step 4**: Some meta information is needed to store with rule, therefore, the meta information is extracted and build *Maintenance Slot*.

- **Step 5**: Extract some information about purpose, explanation of rules, keywords of rule, and citation of rule and build the *Library Slot*.
- **Step 6**: From the plain rule identify and extract condition part.
- **Step 7**: From the plain rule identify and extract conclusion part.
- **Step 8**: The *Data Slot* is prepared based on the concepts used in facts of the condition as input parameters, while the output parameters are generated from conclusion part. The *Data Slot* is generate with emergence of standard concept of terminology and data model using mapping files.
- **Step 9**: The *Logic Slot* is prepared from condition part of the plain rule with amalgamation of the standard terminological concepts and standard data model classes and attributes.
- **Step 10**: The *Action Slot* is built from the conclusion part using standard concepts.
- **Step 11**: Finally, all the slots, like *Maintenance slot*, *Library Slot*, *Knowledge Slot*, *Data Slot*, *Logic Slot*, and *Action Slot* are merged into a single Medical Logic Module (MLM).

5.3 Knowledge Slots Generation

As we mentioned in aforementioned section 5.1 that the *Knowledge Slot* is the core of Medical Logic Module (MLM). It is responsible to provide inferencing capability to knowledge base for generating recommendations. The knowledge rule has two parts condition and conclusion, based on this philosophy, the Knowledge Slot is categorized in three sub-categories to represent *Data*, *Logic*, and *Action*. All these slots are generated based on the condition and conclusion of the rule, each slot is generate based on a particular process. Each slot generation processes are shown in Figure 5.5

5.3.1 Data Slot Generation

The *Data Slot* deals with input data required for the execution of the corresponding MLM, and output data that will be produced as recommendation, decision, and guideline. The Data Slot

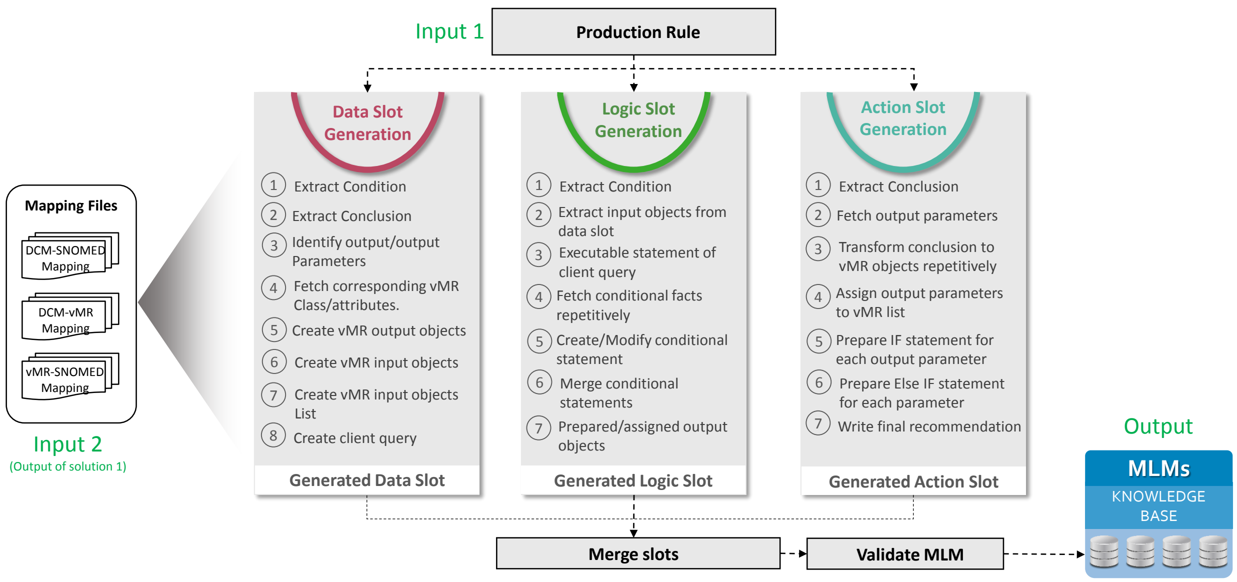

Figure 5.5: Internal processes of slots generation

usually contains input objects, output objects, their corresponding lists of objects, and a query, which will run on the client system to fetch the required data. The *Data Slot* generation has following eight steps process.

- **Step 1**: Extract condition from the plain rule.
- **Step 2**: Extract conclusion from the plain rule.
- **Step 3**: Identify input and output parameters from keys and values in conditional and conclusion facts.
- **Step 4**: Corresponding mapped vMR schema classes and attributes fetch for keys, while corresponding mapped terminological codes for the values in facts.
- **Step 5**: Create output objects individually for each output in form of vMR classes and attributes.
- **Step 6**: Generate vMR input objects individually for each input with vMR classes and attributes.

- **Step 7**: Prepared one combined list of objects for input parameters only.
- **Step 8**: Finally, create the client, which will be executed on the client system to fetch the required data based on the input parameters.

The aforementioned steps of *Data Slot* generation are realized using Algorithm 4.

5.3.2 Logic Slot Generation

The *Logic Slot* is responsible to represent the main logic of the rule, specifically the condition part. It transforms the IF statement of rule to the standard format of shareable knowledge representation, we emerge the standard concepts to the this statement. The IF statement usually contains multiple facts of the condition part, and each fact is the combination of key, values, and operators. All the facts are joint using joint operators such AND and OR operators. The *Logic Slot* generation has following seven steps process.

- **Step 1**: Extract condition from the plain rule.
- **Step 2**: Extract all the input objects from *Data Slot*.
- **Step 3**: It generates the executable statement of the client query to bring required input data from the client's system.
- **Step 4**: In this step, all the conditional facts are fetched from the condition, repetitively.
- **Step 5**: Create and modify the the conditional statement for all facts with the emergence of vMR Data model classes/attributes and SNOMED CT codes for conditional keys and values, respectively.
- **Step 6**: Merge all the transformed conditional facts into a single IF conditional statement.
- **Step 7**: Finally, the output object is populated with possible values set.

The aforementioned logic slot generation steps are shown in Algorithm 5.

Algorithm 4: Semantic Reconciliation Model realization for Data Slot generation

Input : $Rule$ production rule
$mfDCMSNOMED$ mapping file of DCM-SNOMED
$mfvMRSNOMED$ mapping file of vMR-SNOMED
$mfDCMvMR$ mapping file of DCM-vMR

Result: $DataSlot$

Initialization:
inputObjectList = $Empty$;
outputObjectList = $Empty$;
listConditionalOperator[2] = {"and", "or"};
listConditionalFacts = $Rule.condition$.getAllFacts(listConditionalOperator);
listConclusionFacts = $Rule.conclusion$.getAllFacts(listConditionalOperator);
listComparisonOperator[] = $\{" = "," > "," < "," >= "," <= "\}$;
foreach $Fact f$ *of* $listConditionalFacts$ **do**
    operands[2] = f.getOperands(listComparisonOperator);
    foreach $Concept C$ *in* $mfDCMvMR$ **do**
        if C *is mapped with* $operand[1]$ **then**
            $vMRClassAttribute \leftarrow getCorrespondingvMRClass(C)$;
            if *inputObjectList.contains(*$vMRClassAttribute$ *= False)* **then**
                $inputObjectList$.CreateObject($vMRClassAttribute.class$);
                $SNOMEDCode = mfDCMvMR$.getSNOMEDCode(C);
                queryList.createQuery($vMRClassAttribute.class, SNOMEDCode$)
                Break;
            end
        end
    end
end
foreach $Fact f$ *of* $listConclusionFacts$ **do**
    operands[2] = f.getOperands(listComparisonOperator);
    foreach $Concept C$ *in* $mfDCMvMR$ **do**
        if C *is mapped with* $operand[1]$ **then**
            $vMRClassAttribute \leftarrow getCorrespondingvMRClass(C)$;
            if *outputObjectList.contains(*$vMRClassAttribute$ *= False)* **then**
                $outputObjectList$.CreateObject($vMRClassAttribute.class$);
                Break;
            end
        end
    end
end

5.3.3 Action Slot Generation

The *Action Slot* handles the recommendation part of the knowledge rule from the conclusion part. When the condition part of the *Logic Slot* is executed and returned true values then IF statement of *Action Slot* will be executed, otherwise, $Then$ part will be executed. Again, we amalgamate the standard codes and attributes of standard terminology and data model with this slot. The *Action Slot* generation has following seven steps process.

- **Step 1**: Extract conclusion from the plain rule.
- **Step 2**: Fetch all output object, which are generated during *Data Slot*.
- **Step 3**: All the conclusion parts are transformed into corresponding standard codes and assigned to the standard data model attributes of specified classes, repetitively.
- **Step 4**: Assign all the output objects to the list of objects to send in a single packet.
- **Step 5**: In step five, prepare the IF statements for each output object in the list.
- **Step 6**: For each IF statement generated in previous step, it generates the $ELSE$ statement to execute when the condition of *Logic Slot* comes false.
- **Step 7**: Finally, the recommendation or decision is transformed to the objects.

The aforementioned logic slot generation steps are shown in Algorithm 6.

5.4 Summary

In this chapter, we discussed the *Structure Level Semantic Reconciliation* to transform the production rules into shareable and interoperable knowledge. The shareability aspect of the knowledge is achieved by standard representation of knowledge such as Medical Logic Module, while the interoperability is achieved with the help of our previous solution of *Schema-Data Level Semantic Reconciliation*, discussed in previous chapter.

Algorithm 5: Semantic Reconciliation Model realization for Logic Slot generation

```
Input  : Rule production rule
         inputObjectsList Input object list generated by Data Slot
         outputObjectsList Output object list generated by Data Slot
         mfDCMSNOMED mapping file of DCM-SNOMED
         mfvMRSNOMED mapping file of vMR-SNOMED
         mfDCMvMR mapping file of DCM-vMR
Result: LogicSlot
Initialization:conditionStatement = "" and
conclusionStatement = "";
listConditionalOperator[2] = {"and", "or"};
listConditionalFacts = Rule.condition.getAllFacts(listConditionalOperator);
listConclusionFacts = Rule.conclusion.getAllFacts(listConditionalOperator);
listComparisonOperator[] = {" = "," > "," < "," >= "," <= "};
foreach Factf of listConditionalFacts do
    operands[2] = f.getOperands(listComparisonOperator);
    foreach ConceptC in operands do
        if inputObjectsList.find(ConceptC) = True then
            conditionStatement = conditionStatement +
             inputObjectsList[c].operands[0].getVMRClasses() + "." +
             inputObjectsList[c].getVMRAttribute();
            conditionStatement = conditionStatement +
             inputObjectsList[c].operands[0].getSNOMEDCTCode()
             conditionStatement = conditionStatement +
             operands[0].getFollowedOperator();
        end
    end
end
foreach Factf of listConclusionFacts do
    operands[2] = f.getOperands(listComparisonOperator);
    foreach ConceptC in operands do
        if outputObjectsList.find(ConceptC) = True then
            conclusionStatement = conclusionStatement +
             outputObjectsList[c].operands[0].getVMRClasses() + "." +
             outputObjectsList[c].getVMRAttribute();
            conclusionStatement = conclusionStatement +
             outputObjectsList[c].operands[0].getSNOMEDCTCode()
             conclusionStatement = conclusionStatement +
             operands[0].getFollowedOperator();
        end
    end
end
LogicSlot = "IF " + conditionStatement + " Then" + conclusionStatement;
```

Algorithm 6: Semantic Reconciliation Model realization for Logic Slot generation

Input : $Rule$ production rule
$outputObjectsList$ Output object list generated by Logic Slot
Result: $ActionSlot$

Initialization: recommendationStatement = ””;
foreach *Object obj of* $outputObjectsList$ **do**
  if obj*.Is Not Null* **then**
    $recommendationStatement$ = ”IF recommendation =” + obj.getVMRClass() + ” then ”;
    $recommendationStatement$ = $recommendationStatement$ + ”Write ” + obj.getVMRClass().getVMRAttribute() = obj.getValue;
    ”Write ” + $recommendationStatement$;
  end
end
if $recommendationStatement$ *is not NULL* **then**
  $LogicSlot$ = $recommendationStatement$;
end

Chapter 6

Results and Evaluation

The SRM provides two types of reconciliation model *Schema-Data Level Semantic Reconciliation* and *Structure Level Semantic Reconciliation*, therefore, we divided the results into two following sections.

6.1 Results: Schema-Data Level Semantic Reconciliation

The *Schema-Data Level Semantic Reconciliation* provides multi-model mapping among standard and non-standard terminologies, data model, and localized ontology. We evaluated our proposed system of multi-model mapping for two mapping methodologies, *a) DCM-Standard Terminology Mapping*, and b) *Standard Terminology and Data Model Mapping*.

6.1.1 Results: DCM-Standard Terminology Mapping

We evaluated the system using two datasets SNOMED CT ontology downloaded from the website of the International Health Terminology Standard Development Organization (IHTSDO) [108] and DCM local terminology developed for our collaborative hospital. The SNOMED CT contains more than 0.3 million concepts, 200 properties, maximum depth 9, and maximum number of children is 13, while the DCM comprises 214 concepts and seven properties of head and neck cancer domain. The results' statistics are shown in Table 6.1. We calculated the precision, recall, and F-measure using corresponding standard formulas based on the values described in Table 6.1. We measured the precision, recall, and F-measure as shown in Figure 6.1. The objective of our study is to achieve high precision and recall. The precision is highly affected by the regional concepts and some non-standard acronyms used in DCM. Some regional concepts related to drugs such as "naswar" and "paan", which only use in the specific region of our collaborative hospital.

Table 6.1: MAPPING STATISTICS OF SNOMED CT AND DCM.

Total DCM Cloncepts	214
SNOMED CT Overall Concepts	0.3 Million concepts
Mapped concepts with SNOMED CT	197
Wrong Mapped Concepts	9
Local Concepts	7

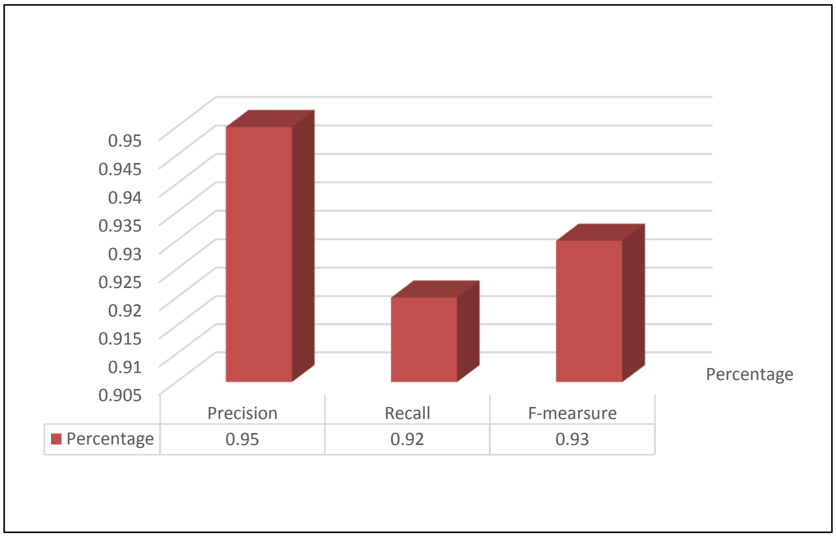

Figure 6.1: DCM and SNOMED CT mapping results

Similarly, some non-standard acronyms such as “S Proc 1” and “C S RT”, are used in DCM by local physicians. Therefore, the non-standard acronyms and regional concepts do not exist in standard SNOMED CT terminology and it affected the precision and recall of the system.

We evaluate our system with state-of-the-art systems of ontology matching techniques based on their participation in Ontology Alignment Evaluation Initiative (OAEI) [46]. The evaluation systems comprises AgreementMakerLight (AML) [41], GOMMA [43], and LogMap Light [42]. We measured the precision, recall, and F-measure of all these systems using the following formulas (Equations 6.1, 6.2, and 6.3), respectively, and compared with existing systems, as a result, our proposed system’s precision, recall, and F-measure are better than the existing systems. Table 6.2 shows the confusion matrix of existing systems with our proposed system, and graphically the

Table 6.2: Confusion matrix of the existing systems for DCM-SNOMED Mapping.

Systems	True Positive	False Positive	Local Concepts	False Negative
AgreementMakerLight	163	43	8	51
GOMMA	137	69	8	77
LogMap Light	153	53	8	61
Proposed System	197	9	8	17

comparison is shown in Figure 6.2.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{6.1}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{6.2}$$

$$F - Measure = 2X\frac{Precision.Recall}{Precision + Recall} \tag{6.3}$$

6.1.2 Results: Standard Terminology and Data Model Mapping

In this section, we evaluated the performance of the proposed definition based *Standard Terminology and Data Model Mapping* algorithm for matching the standard data model (vMR) and standard terminology (SNOMED CT) concepts. First, we collect all definitions from both vMR standard data model and SNOMED CT standard terminology as a dataset for the experiments. The SNOMED CT contains more than 0.3 million concepts, 200 properties, maximum depth 9, and maximum number of children is 13, but we considered the top hierarchical 21 concepts with definitions. Similarly, the standard data model vMR has 94 classes with 335 properties and we considered 69 classess with 171 attributes and their definitions for the experiment, which are specifically recommended for CDSS systems. For evaluation propose, we compared and marked all vRM with SNOMED CT definitions manually. In this experiment, the proposed definition base

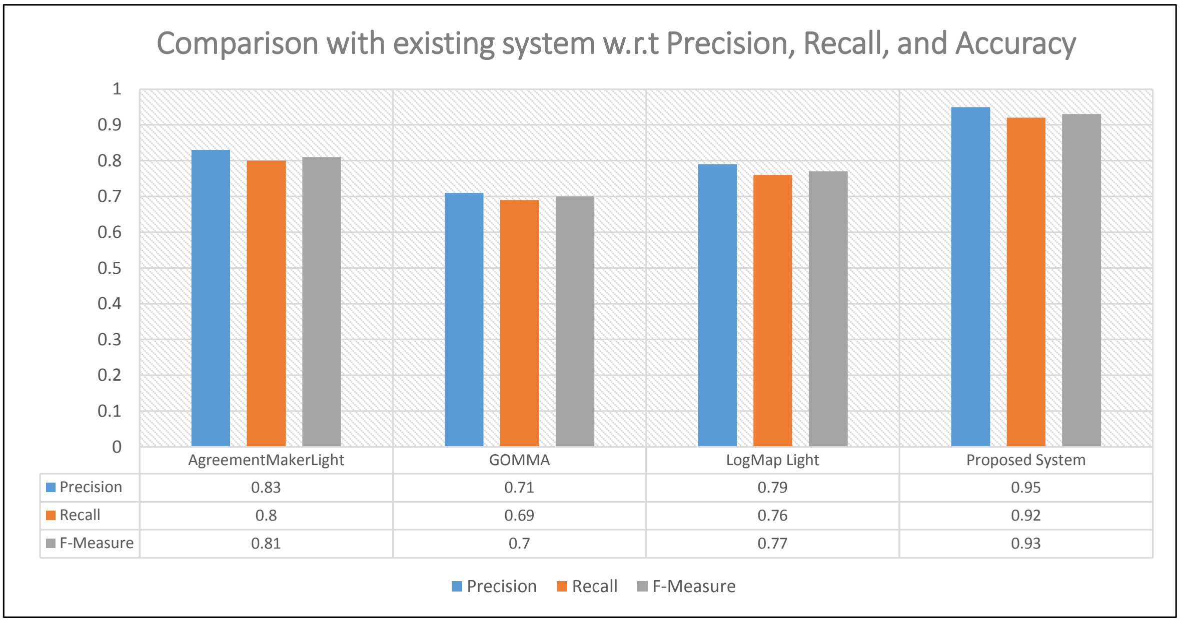

Figure 6.2: DCM and SNOMED CT mapping comparison with existing systems

matching algorithm was evaluated and compared with the base-line algorithm (Jaccard similarity) based on the accuracy and efficiency of the algorithm. The accuracy refers to the number of exact matches as manually labeled and the efficiency refers to the number of comparisons performed to detect those exact matches. Our proposed algorithm is mainly based on the space vector creation before performing matching. The space vector is generated by applying the text processing steps such as tokenizing the definitions, case transformation, filter stop words. Each filtered token is expended by applying the *WordNet* and *ConceptNet5* synonyms, hyponym, and hypernyms as explicit semantics. Additionally, we employed the UMLS dictionary for the concepts and entity detection to enhance the space vector for improving the matching performance. We also added a lexical chain of implicit semantics into the space vectors from *ConceptNet5*. The experiments were performed using data analytics-RapidMiner with text processing extension. The evaluation is performed with different threshold value ranging from 0.55 to 0.75 with the gap of 0.05. The Results of both base-line Jaccard similarity and proposed Algorithms are shown in Table 6.3 and Table 6.4, respectively.

Table 6.3: Definition Based Matching with base-line (Jaccard Similarity) algorithm.

Threshold Values	Recall	Precision	F-Score	True Positive	False Positive	False Negative
0.55	0.92	0.59	0.72	65	45	5
0.6	0.85	0.77	0.81	70	20	12
0.65	0.78	0.8	0.79	73	18	20
0.7	0.75	0.85	0.8	72	12	23
0.75	0.82	0.88	0.76	80	10	25

Table 6.4: Definition based mapping with proposed Standard Terminology and Data Model Mapping algorithm.

Threshold Values	Recall	Precision	F-Score	True Positive	False Positive	False Negative
0.55	0.97	0.66	0.78	75	38	2
0.6	0.96	0.79	0.87	75	19	3
0.65	0.95	0.86	0.9	78	12	4
0.7	0.96	0.89	0.92	81	10	3
0.75	0.97	0.89	0.93	85	10	2

The results show the most accurate results of both the algorithms base-line (Jaccard Similarity) and proposed system at threshold value 0.75. The proposed algorithm mapped standard data model with standard terminology with better precision, recall, and F-Measure than the base-line Jaccard similarity algorithm. The highest score of our proposed algorithm at threshold value 0.75 are precision (0.89), recall (0.97), and F-measure (0.93) as shown in Figure 6.3, which is better than the base-line (Jaccard similarity) results as precision (0.88), recall (0.82), and F-measure (0.76) as shown in Figure 6.4. In the comparison of our proposed definition based algorithm for *Standard Terminology and Data Model Mapping* and the base-line (Jaccard similarity) algorithm, we compared the F-measure, and showed in Figure 6.5

6.2 Results: Structure Level Semantic Reconciliation

6.2.1 Case Study: Treatment Plans for Oral Cavity Lesions

We selected the formally extracted refined-clinical knowledge model (RCKM) from our previous work [2]. In this work, we used data-driven knowledge acquisition for real SKMCH patient data to generate a predictive model (PM). The PM was attained using a decision tree algorithm, chi-square automatic interaction detection (CHAID), on the dataset of 1229 patients. Simultaneously,

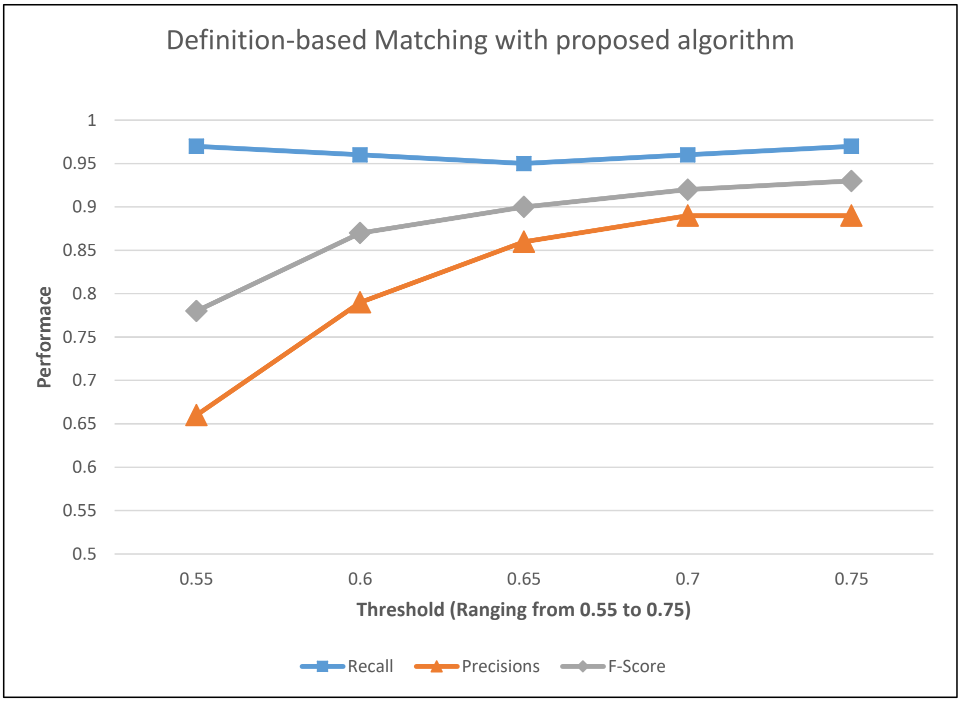

Figure 6.3: Standard Terminology and Data Model mappings using proposed algorithm

a team of physicians created a clinical knowledge model (CKM) for the oral cavity site of head and neck cancer from a well-known online resource, National Comprehensive Cancer Network (NCCN) guidelines [116]. Finally, the R-CKM was created by a rigorous validation process of conforming the PM as a final model to the CKM. In this study, the created R-CKM specifically focuses on treatment plans for head and neck cancer with emphasis on the oral cavity, as shown in Figure 6.6. For a given R-CKM tree, a set of eight rules can be created based on decision nodes for recommended treatment plans, as shown in Table 6.5. We created a single MLM for each corresponding rule and integrated the compiled version into the HMIS system. In this scenario, we focus on a single MLM for Rule 5, with the following steps performed in creating this rule.

Step 1: We display all required information about the MLM on the *Rule Editor* screen such as *Rule Title*, *MLM Name*, *Citation*, *Purpose*, and *Explanation*. *Author's name*, *Institution*, and *Created Date* appear by default from the author's profile information. The detailed implementation of the authoring environment is provided in A.1 with complete features of

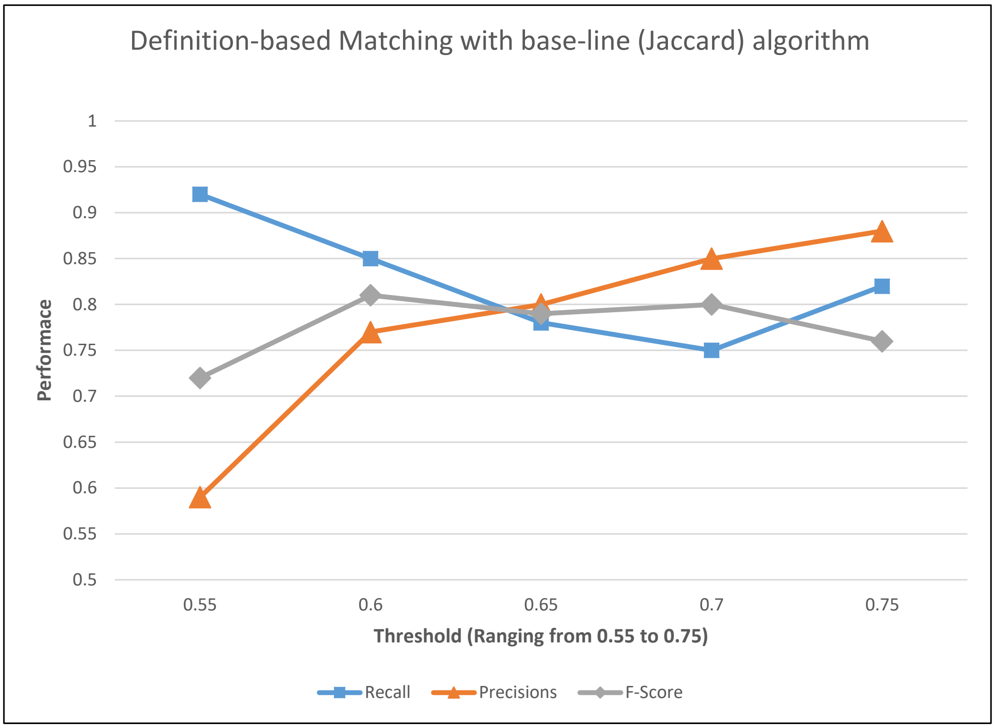

Figure 6.4: Standard Terminology and Data Model mappings using base-line (Jaccard similarity) algorithm algorithm

the system.

Step 2: In the *Condition* box, we write the condition part of the rule. The *Treatment Intent* concept is reflected in the *Condition* box when we select the *Treatment Intent* from the DCM concepts tree, as shown in the highlighted Area 3 of Figure A.2. We write the 'equal to'(=) sign and the IntelliSense window appears with the possible values set for *Treatment Intent*. Here we select the value *Radical*. We write all other inputs in the same manner such as *"Treatment Plan Given = Chemo induction"* with the help of the IntelliSense window and DCM Tree. The condition part in the *Condition* box with IntelliSense functionality is shown in Figure A.3.

Step 3: We follow the procedure in Step 2 for the action part of the rule in the *Action* box of the *Rule Editor*. The *Condition* and *Action* parts after rule completion are shown in Figure 6.7.

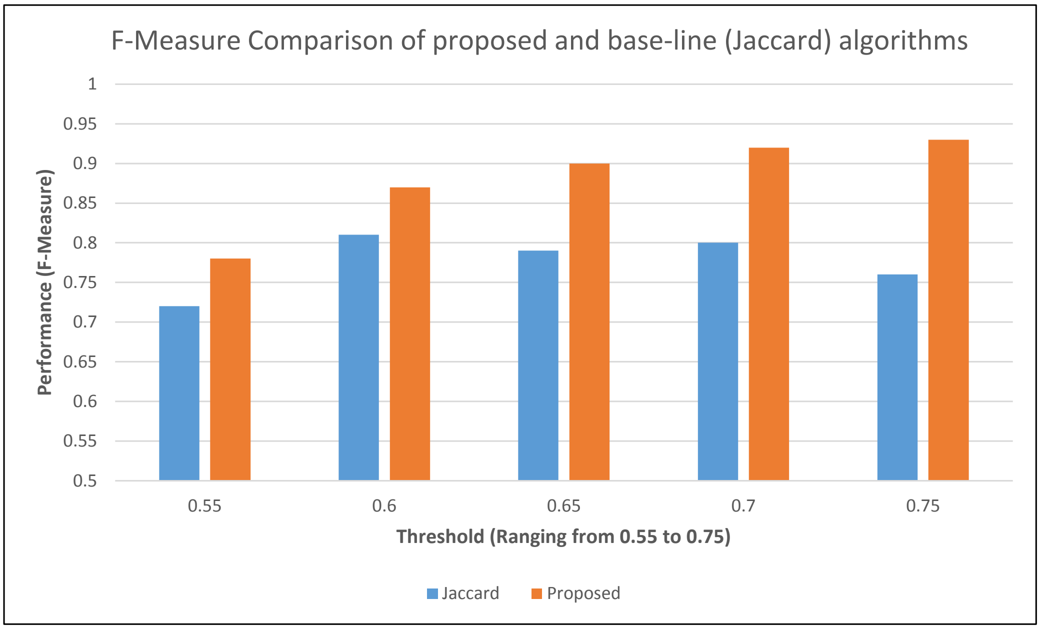

Figure 6.5: F-Measure Comparison of proposed and Jaccard similarity algorithms

Step 4: The rule is saved by pressing the *Save Rule* button.

Step 5: The created MLM can be seen on the *Rule Viewer* screen by clicking the *Show Created MLM* button. The application view of the created MLM is shown in Figure A.4. Moreover, the MLM details can be found in Listing A.1 in Appendix A.2; while the DCM, vMR, and SNOMED CT mappings used to create the MLM are listed in Table 4.1, **??**.

6.2.2 MLMs Validation Using Real Patient Cases

We implemented and validated the proposed system using a real practice dataset from SKMCH. The experimental setup and implementation are as follows.

- We created MLMs from eight rules, shown in Table 6.5, which are modeled from R-CKM as described in Section 6.2.1. This model was initially validated on a real practice dataset of 739 SKMCH patients with model accuracy of 53% [2]. We re-evaluated the R-CKM on recently generated data from 1,783 patients with model accuracy of 73.7%. The R-CKM accuracy ($R - CKM_{acc}$) based on the newly created MLMs is a weighted mean accuracy

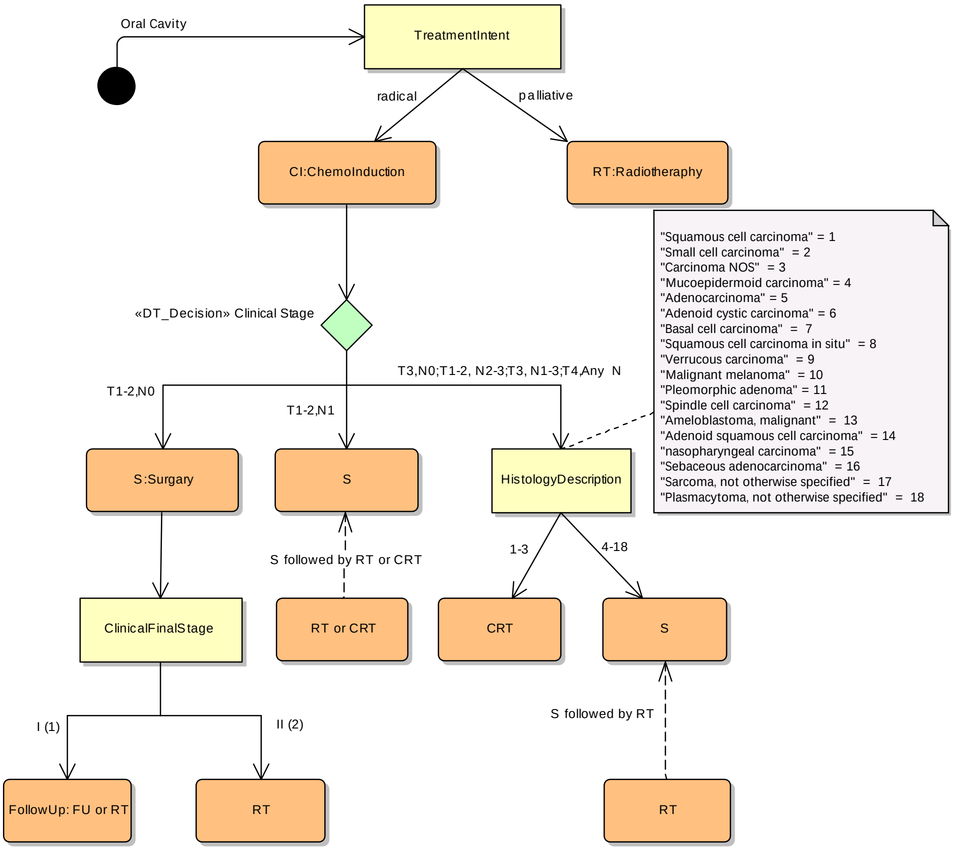

Figure 6.6: Refined clinical knowledge model of a treatment plan for an oral cavity lesion [2].

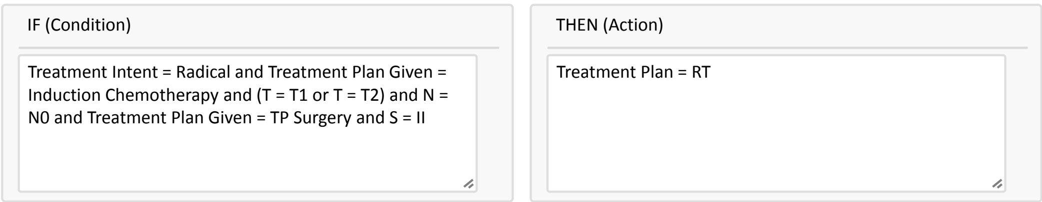

Figure 6.7: Logic component of Rule 5.

of disjoint MLMs calculated by Equation 6.4.

$$R - CKM_{acc} = \frac{\sum_{i=1}^{n}(pat_{MLM_i} \times A_{MLM_i})}{pat_c} \tag{6.4}$$

Where pat_{MLM_i} and A_{MLM_i} represent the number of patient cases assigned to MLM_i and its accuracy, respectively. pat_c represents total patient cases assigned to MLMs: MLM_1 to

Table 6.5: Rules for treatment plan guidelines.

Rule ID	Rule Conditions	Rule Conclusion
Rule 1	Treatment Intent = Palliative	Treatment Plan = Radiotherapy (RT)
Rule 2	Treatment Intent = Radical	Treatment Plan = Chemoinduction
Rule 3	Treatment Intent = Radical and Treatment Plan Given = Chemoinduction and (T = T1 or T = T2) and N = N0	Treatment Plan = Surgery
Rule 4	Treatment Intent = Radical and and Treatment Plan Given = Chemoinduction (T = T1 or T = T2) and N = N0 and Treatment Plan Given = Surgery and S = I	Treatment Plan = RT or Next Followup
Rule 5	Treatment Intent = Radical and Treatment Plan Given = Chemoinduction and (T = T1 or T = T2) and N = N0 and Treatment Plan Given = Surgery and S = II	Treatment Plan = RT
Rule 6	Treatment Intent = Radical and Treatment Plan Given = Chemoinduction and(T = T1 or T = T2) and N = N1	Treatment Plan = Surgery Followed by RT or Chemo-radiotherapy (CRT)
Rule 7	Treatment Intent = Radical and Treatment Plan Given = Chemoinduction and ((T = T3 and N = N0) or ((T = T1 or T2) and (N = N2 or N3)) or (T = T3 and (N = N1 or N2 or N3)) or (T = T4 and N = Any N)) and (Histology = 1 or 2 or 3)	Treatment Plan = CRT
Rule 8	Treatment Intent = Radical and Treatment Plan Given = Chemoinduction and ((T = T3 and N = N0) or ((T = T1 or T2) and (N = N2 or N3)) or (T = T3 and (N = N1 or N2 or N3)) or (T = T4 and N = Any N)) and (Histology = 4 or 5 or 6 or 7 or 8 or 9 or 10 or 11 or 12 or 13 or 14 or 15 or 16 or 17 or 18)	Treatment Plan = Surgery Followed by RT

MLM_8.

Individual MLM accuracy A_{MLM} is calculated as in Equation 6.5.

$$A_{MLM} = \frac{pat_{MLM_c} \times 100}{pat_{MLM}} \tag{6.5}$$

Where pat_{MLM_c} and pat_{MLM} represent the number of correctly classified patient cases by MLM and total patient cases assigned to given MLM, respectively.

- The MLMs generated by our proposed system are developed and deployed as Smart CDSS XML-based web service, which is designed according to the framework mentioned in [8].
- We developed a client application in the .NET environment using C# language that extracts oral cavity cancer patient data from the SKMCH database. The client interacts with the Smart CDSS Service and iteratively launches individual patient data for recommendation. Individual patient cases with associated recommendations are saved in a CSV (comma separated values) file for MLM result verification.

- We tested and validated the MLMs on 1,314 patient cases with 100% correct recommendations for all patients. This evaluation shows that the MLMs generated by our system are error free and do not affect R-CKM accuracy. Table 6.6 describes the distribution of patient cases over individual MLMs.

Table 6.6: Distribution of patient cases.

MLM ID	**Associated Rule ID**	**Contents/Logic Complexity [No. of attributes, {No. of logical operators}]**	**Candidate patient cases**
MLM1	Rule 1	[1, {And (0), Or (0)}]	241
MLM2	Rule 2	[1, {And (0), Or (0)}]	39
MLM3	Rule 3	[5, {And (3), Or (1)}]	121
MLM4	Rule 4	[7, {And (5), Or (1) }]	128
MLM5	Rule 5	[7, {And (5), Or (1)}]	158
MLM6	Rule 6	[5, {And (3), Or (1)}]	99
MLM7	Rule 7	[17, {And (7), Or (9)}]	427
MLM8	Rule 8	[29, {And (7), Or (21)}]	31
Total			**1314**

6.2.3 System Comparison and Evaluation

We evaluated our system by applying system-centric and user-centric evaluations [117, 118]. In the system-centric evaluation, the system was evaluated against a predefined ground truth dataset of opinions. In the user-centric evaluation, the system was evaluated by user interaction with the system based on performance with respect to MLM creation time of MLM.

6.2.3.1 System-centric Evaluation

In the system-centric evaluation, we formulated the results based on the set of requirements for clinical information modeling tools developed by Moreno-Conde et al. in [24]. These requirements were produced after rigorous and intensive surveys and interviews with experts and were categorized into *Essential*, *Recommended*, and *Optional* categories. The total requirements in *Essential*, *Recommended*, and *Optional* categories are 20, 21, and 15, respectively.

A team of knowledge engineers and domain experts was created to review and select the candidate requirements of the Clinical Information Modeling Tool (CIMT) [24] for the knowledge acquisition tools. The team formalized a four-phase model process: *reduction phase*, *enhancement phase*, *interpretation phase*, and *evaluation phase*. The objective of this process was to remove all requirements not directly applicable to the knowledge acquisition tools, to incorporate new appropriate requirements, and to interpret the CIMT-based requirements for the knowledge acquisition tools for evaluation. Finally, our proposed system was evaluated with the existing ArdenSuite tool of Medexter [44, 45] based on the selected requirements.

Reduction Phase: We reduced the total number of requirements by removing the requirements that specifically belonged to CIMT [24] and technology-oriented requirements that were not applicable to knowledge acquisition tools. *Essential* requirements (R) were reduced from 20 to 16 by removing R7, R12, R15, and R20; *Recommended* requirements were reduced from 21 to 16 by removing R24, R28, R35, R36, and R38; and *Optional* requirements were reduced from 15 to 5 by removing R22, R44, R46, R47, R49, R50, and R53-R56 (Figure 6.8).

Enhancement Phase: We added two new requirements to the *Recommended* category based on our experiences and observations from our previous work [18] with SKMCH physicians. The first new requirement, "*Provide Domain Clinical Model in hierarchical form for easy selection of required concepts during knowledge creation,*" was added as extended requirement ER57. The second requirement, "*Knowledge editor should provide the facility of contextual selection of required value of a concept from the values set using the IntelliSense window,*" was added as extended requirement ER58. Both requirements help experts recall domain concepts during knowledge creation. They also reduce the chance of errors in the knowledge base rules by minimizing the likelihood of wrong concept usage. In total, the *enhancement* phase increased the number of *Recommended* requirements from 16 to 18, as shown in Figure 6.8.

Interpretation Phase: We interpreted the consensus requirements of CIMT for knowledge authoring tools that are closely related to CIMP [24]. All clinical knowledge management tools and repositories are highly recommended to follow these requirements in the corresponding tools. We interpreted each requirement R in the final requirement set produced in the *enhancement* phase as the corresponding interpreted requirement (IR), as shown in the column for *Interpretation for Knowledge Authoring Tool* in A.3. The final requirements list after performing the four-phase process is shown in A.3.

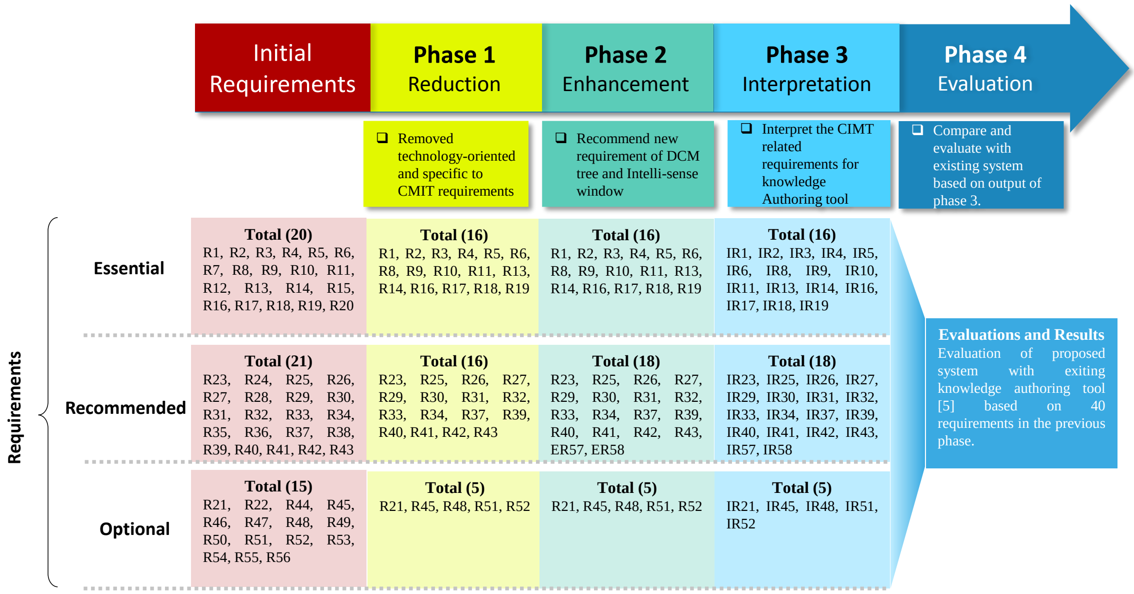

Figure 6.8: Phases for evaluation of I-KAT.

Evaluation Phase: We compared our system with the commercially available ArdenSuite [44, 45] based on the final requirements produced in the *interpretation* phase listed in A.3. A detailed comparison based on the final requirements list is shown in Table 6.7. We classified the implementation of requirements into three categories: *fully supported* (FS), *partially supported* (PS), and *not-supported* (NS). A requirement is *fully supported* when the system has implemented it; if the system has not implemented it, then it is considered as *not-supported*. If some part of the requirement is implemented or has partial functionality, then it is considered as *partially supported*. For instance in IR9, ArdenSuite [44, 45] validates the knowledge rule syntax according to the standard MLM syntax, while rule creation semantics depend on expert knowledge. The comparison list in Table 6.7 shows the priority, requirement number, and implementation status of the requirements with a tick mark (✓) in the corresponding implementation category for each tool.

Table 6.7: **Comparison table of I-KAT and ArdenSuite with respect to implementation category [Essential: E, Recommended: R, Optional: 0].**

Priority	Req. Number	I-KAT			ArdenSuite		
		NS	PS	FS	NS	PS	FS
E	IR1			✓		✓	
E	IR2			✓		✓	
E	IR3			✓	✓		
E	IR4			✓			✓
E	IR5			✓			✓
E	IR6			✓			✓
E	IR8			✓			✓
E	IR9			✓		✓	
E	IR10			✓	✓		
E	IR11			✓			✓
E	IR13			✓			✓
E	IR14			✓			✓
E	IR16			✓	✓		
E	IR17	✓			✓		
E	IR18			✓		✓	
E	IR19			✓			✓
R	IR23			✓			✓
R	IR25			✓		✓	
R	IR26			✓	✓		
R	IR27	✓					✓
R	IR29		✓		✓		
R	IR30			✓		✓	
R	IR31		✓				✓
R	IR32			✓	✓		
R	IR33			✓			✓
R	IR34		✓		✓		
R	IR37			✓	✓		
R	IR39			✓		✓	
R	IR40			✓		✓	
R	IR41	✓					✓

Continued on next page

Table 6.7 – *Continued from previous page*

Priority	Req. Number	I-KAT			ArdenSuite		
		NS	PS	FS	NS	PS	FS
R	IR42			✓		✓	
R	IR43			✓	✓		
R	ER57			✓	✓		
R	ER58			✓	✓		
O	IR21			✓		✓	
O	IR45			✓	✓		
O	IR48			✓	✓		
O	IR51			✓			✓
O	IR52	✓				✓	

The comparison performed in the *evaluation* phase is graphically depicted in Figure 6.9. The graph shows that I-KAT provides full support to 32 out of 39 (82.05%) requirements, partial support to 3 out of 39 (7.69%) requirements, and no support to the remaining 4 out of 39 (10.25%) requirements. In contrast, ArdenSuite provides full support to 14 (35.89%) requirements, partial support to 11 (28.20%) requirements, and no support to 14 (35.89%) requirements. This shows that I-KAT offers higher implementation support for the requirements than ArdenSuite.

All three implementation categories (i.e., FS, PS, and NS) are inversely proportional to each other. Therefore, I-KAT has a higher percentage of implemented requirements in FS and a relatively low percentage in PS and NS compared to ArdenSuite. Figure 6.10 shows the detailed individual graphs of the comparison between I-KAT and ArdenSuite with respect to implementation categories for all requirement categories. Figure 6.10(a) depicts that I-KAT provides no support for one *Essential*, two *Recommended*, and one *Optional* requirements, while ArdenSuite provides no support for four, eight, and two requirements, respectively. I-KAT provides *partial* support for three *Recommended* requirements only, while ArdenSuite has partial support for four *Essential*, five *Recommended*, and two *Optional* requirements, as shown in Figure 6.10(b). I-KAT provides full support for 15 out of 16 Essential requirements, 13 out of 18 Recommended, and 4 out of 5 Optional requirements, as shown in Figure 11 (c). On the other hand, ArdenSuite sup-

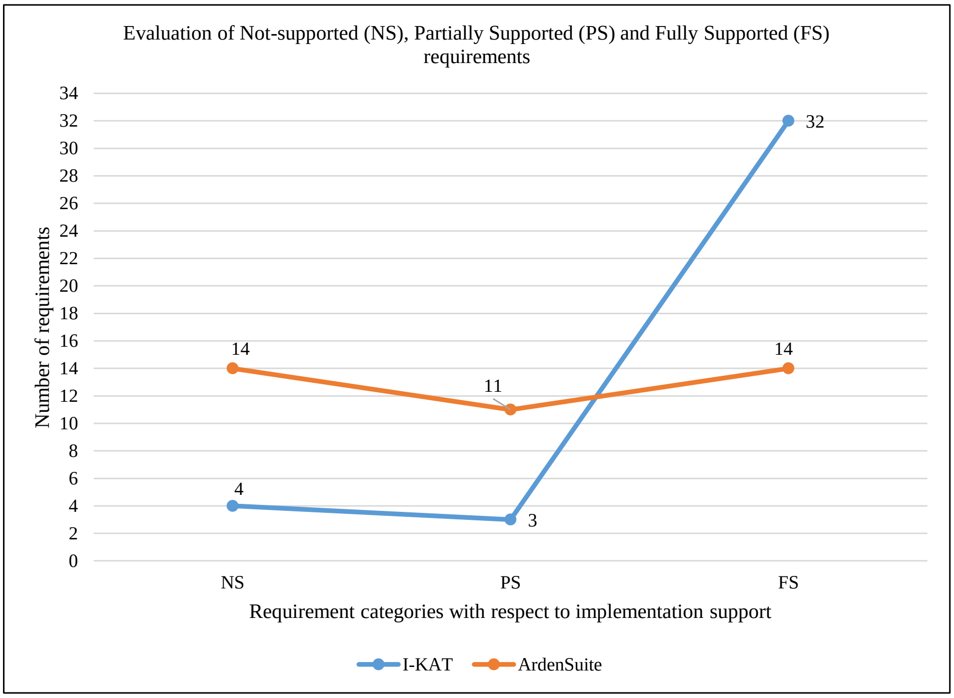

Figure 6.9: Accumulative comparison of I-KAT and ArdenSuite with respect to NS, PS, and FS implementation requirements.

ports 8 out of 16 *Essential*, 5 out of 18 *Recommended*, and 1 out of 5 *Optional* requirements. This overall evaluation shows that I-KAT exhibits higher implementation support for the requirements in all three categories.

6.2.3.2 User-centric Evaluation

The main focus of our proposed system was to create an easy-to-use interface for creating shareable MLMs. A system with an easy-to-use interface is more time efficient than complex systems that require a great deal of time to produce the required results. Therefore, we considered time when evaluating the user friendliness of our system. Our second objective was to generate sharable knowledge with minimal complexity for physicians; therefore, we selected MLM validation as the second criterion for evaluation. In MLM validation with respect to errors, we focused on syntax and structure complexity as well as the accuracy of vMR classes, attributes, and SNOMED CT codes incorporated in the created MLM.

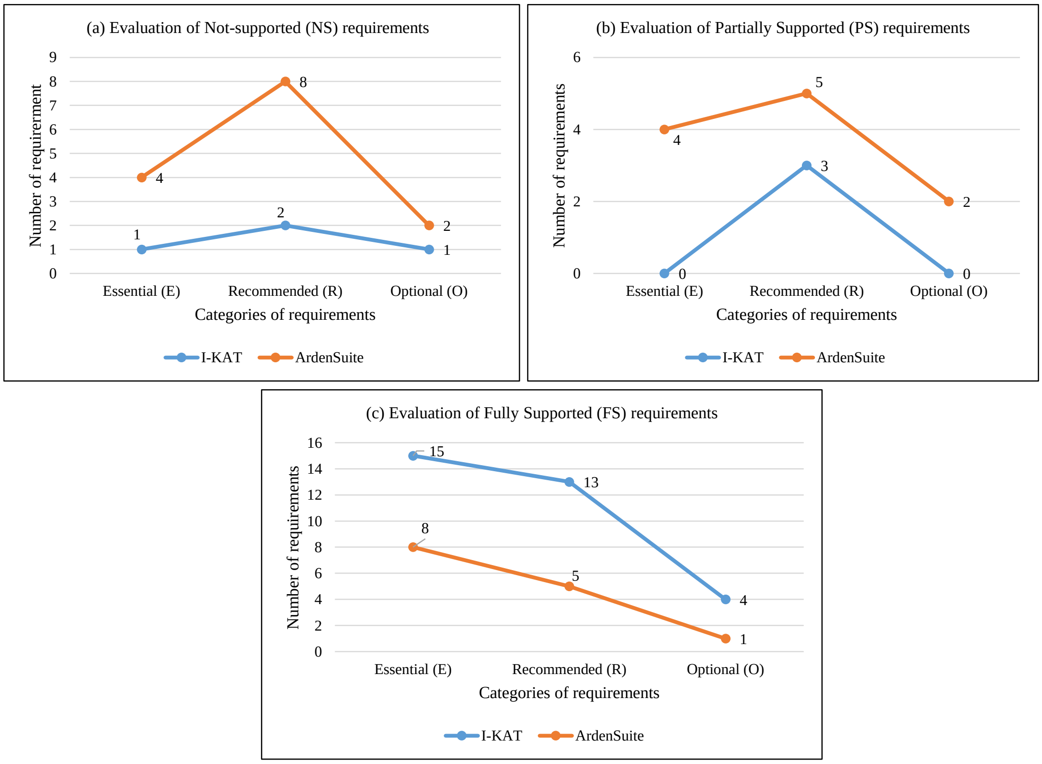

Figure 6.10: Individual comparison of I-KAT and ArdenSuite with respect to requirement categories: Essential (E), Recommended (R), and Optional (O).

We evaluated our system with a knowledge engineer, physicians with Arden Syntax experience, and a physician with no such experience. In our experiment, three physicians and one knowledge engineer (total = 4) participated with the following expertise levels in Arden Syntax.

- Physician 1: Experienced
- Physician 2: Intermediate
- Physician 3: Novice
- Knowledge Engineer : Experienced

As a prerequisite, we provided the complete mappings discussed in the *Semantic Reconciliation Model* (SRM) section and trained the participants using basic artifacts of HL7 Arden Syntax for MLM creation in ArdenSuite [44, 45] using these mappings. We also provided the Arden Syntax specification to participants. In the experiment, participants created MLMs for each rule described in Table 6.5 built from the guidelines of the *Treatment Plans* for *Oral Cavity*, as discussed in the previous scenario. In the first session, each participant created MLMs for Rule 1, Rule 3, Rule 5, and Rule 7 based on contents and logic complexity, as shown in Table 6.6. Each participant used ArdenSuite per our instructions to create an MLM and then created the same MLM using our proposed system. In the second session, we switched the sequence of rule editors and rules based on content and logic complexity, as shown in Table 6.6. Participants created MLMs for Rule 2, Rule 4, Rule 6, and Rule 8 to avoid bias when using our proposed system initially and then creating the same MLM using ArdenSuite. The experiments showed the following results.

Ease-of-use Evaluation Our proposed system enhanced participants' average performance by a factor of 34 for the simplest MLM for Rule 1 creation and by 5 for the complex MLM for Rule 8 creation. The overall average performance showed a 15-fold improvement. Table 6.8 lists the time in which participants performed the tasks.

MLM Validation Comparison With Respect to Errors: In the MLM validation, we recorded the logical and syntactic errors that occurred during MLM creation. For syntactic errors, we considered errors like missing semicolons, missing variable declaration, and missing colon and equal signs in the assignment operator. For logical errors, we considered incorrect vMR concepts, logical IF constructions, and incorrect use of logical operators. Using ArdenSuite, the participants made on average of 4, 3, 15, 16, 15, 14, 17, and 17 errors (syntactic or/and logical) for MLM1 to MLM8, respectively. The average number of errors made during MLM creation using ArdenSuite was 13. Using our proposed system, the average number of errors made during MLM creation was 1. There were no syntax errors in MLMs created by our system because the syntax complexity is hidden from the physicians. The logical errors made by participants when using our system occurred due to incorrect selection from the DCM concepts tree or IntelliSense window. The syntactic and logical errors made during the experiment are shown in Table 6.9.

Table 6.8: Ease-of-use evaluation with respect to time.

MLM No	**MLM Creation Time**		**User Involved**
	Using ArdenSuite	**Using I-KAT**	
MLM1	18 min 20 Sec	22 sec	Physician 1
	21 min 15 Sec	46 sec	Physician 2
	Not Applicable	66 sec	Physician 3
	8 min 30 Sec	20 sec	Knowledge Engineer
MLM2	18 min 22 Sec	23 sec	Physician 1
	21 min 10 Sec	40 sec	Physician 2
	Not Applicable	69 sec	Physician 3
	8 min 34 Sec	23 sec	Knowledge Engineer
MLM3	32 min 20 Sec	2 min and 47 sec	Physician 1
	34 min 30 Sec	3 min and 5 sec	Physician 2
	Not Applicable	2 min and 40 sec	Physician 3
	19 min 15 Sec	2 min and 18 sec	Knowledge Engineer
MLM4	33 min 25 Sec	3 min and 49 sec	Physician 1
	35 min 39 Sec	3 min and 7 sec	Physician 2
	Not Applicable	3 min and 45 sec	Physician 3
	18 min 21 Sec	2 min and 19 sec	Knowledge Engineer
MLM5	33 min 25 Sec	3 min and 49 sec	Physician 1
	35 min 39 Sec	3 min and 7 sec	Physician 2
	Not Applicable	4 min and 47 sec	Physician 3
	21 min 21 Sec	3 min and 19 sec	Knowledge Engineer
MLM6	32 min 20 Sec	2 min and 47 sec	Physician 1
	34 min 30 Sec	3 min and 5 sec	Physician 2
	Not Applicable	2 min and 40 sec	Physician 3
	19 min 15 Sec	2 min and 18 sec	Knowledge Engineer
MLM7	34 min 45 Sec	4 min and 53 sec	Physician 1
	36 min 51 Sec	5 min and 51 sec	Physician 2
	Not Applicable	8 min and 27 sec	Physician 3
	21 min 34 Sec	4 min and 10 sec	Knowledge Engineer
MLM8	35 min 58 Sec	5 min and 23 sec	Physician 1
	37 min 51 Sec	6 min and 19 sec	Physician 2
	Not Applicable	9 min and 47 sec	Physician 3
	22 min 46 Sec	5 min and 13 sec	Knowledge Engineer

Table 6.9: MLM syntactic and semantic evaluation.

MLM No	MLM Errors Recorded (L: Logical errors, S: Syntax error		User Involved
	Using ArdenSuite	Using I-KAT	
MLM1	S:2, L:2	S:0, L:0	Physician 1
	S:3, L:5	S:0, L:0	Physician 2
	Not Applicable	S:0, L:0	Physician 3
	S:0, L:0	S:0, L:0	Knowledge Engineer
MLM2	S:2, L:1	S:0, L:0	Physician 1
	S:3, L:4	S:0, L:0	Physician 2
	Not Applicable	S:0, L:1	Physician 3
	S:0, L:0	S:0, L:0	Knowledge Engineer
MLM3	S:10, L:11	S:0, L:1	Physician 1
	S:5, L:18	S:0, L:1	Physician 2
	Not Applicable	S:0, L:2	Physician 3
	S:2, L:0	S:0, L:0	Knowledge Engineer
MLM4	S:9, L:13	S:0, L:1	Physician 1
	S:6, L:17	S:0, L:2	Physician 2
	Not Applicable	S:0, L:2	Physician 3
	S:3, L:0	S:0, L:1	Knowledge Engineer
MLM5	S:7, L:12	S:0, L:1	Physician 1
	S:6, L:16	S:0, L:1	Physician 2
	Not Applicable	S:0, L:3	Physician 3
	S:3, L:1	S:0, L:0	Knowledge Engineer
MLM6	S:8, L:9	S:0, L:1	Physician 1
	S:6, L:17	S:0, L:2	Physician 2
	Not Applicable	S:0, L:1	Physician 3
	S:1, L:0	S:0, L:0	Knowledge Engineer
MLM7	S:8, L:13	S:0, L:1	Physician 1
	S:8, L:17	S:0, L:2	Physician 2
	Not Applicable	S:0, L:4	Physician 3
	S:3, L:2	S:0, L:0	Knowledge Engineer
MLM8	S:9, L:15	S:0, L:2	Physician 1
	S:6, L:16	S:0, L:1	Physician 2
	Not Applicable	S:0, L:3	Physician 3
	S:2, L:3	S:0, L:0	Knowledge Engineer

Table 6.10: I-KAT: Time-On-Task data for 4 participants and 8 MLMs.

Participants	MLM1	MLM2	MLM3	MLM4	MLM5	MLM6	MLM7	MLM8
Physician 1	22	23	167	229	229	167	293	323
Physician 2	46	40	185	187	187	185	351	379
Physician 3	66	69	160	225	287	160	507	587
Knowledge Engineer	20	23	138	139	199	138	250	313
Mean	38.5	38.75	162.5	195	225.5	162.5	350.25	400.5
Median	34	31.5	163.5	206	214	163.5	322	351
Geometric Mean	33.99	34.76	161.61	191.30	222.38	161.61	337.89	387.26
Standard Deviation	21.80	21.70	19.43	41.85	44.64	19.43	112.39	127.68
Confidence	21.37	21.26	19.04	41.01	43.74	19.04	110.14	125.12
Confidence Interval (+)	59.87	60.01	181.54	236.01	269.24	181.54	460.39	525.62
Confidence Interval (-)	17.12	17.48	143.45	153.98	181.75	143.45	240.10	275.37
Confidence Interval	42.74	42.53	38.08	82.03	87.49	38.08	220.29	250.24

Table 6.11: ArdenSuite: Time-On-Task data for 4 participants and 8 MLMs.

Participants	MLM1	MLM2	MLM3	MLM4	MLM5	MLM6	MLM7	MLM8
Physician 1	1100	1102	1940	2005	2005	1940	2085	2158
Physician 2	1275	1270	2070	2139	2139	2070	2211	2271
Physician 3	3600	3600	3600	3600	3600	3600	3600	3600
Knowledge Engineer	510	514	1155	1101	1281	1155	1294	1366
Mean	1621.25	1621.5	2191.25	2211.25	2256.25	2191.25	2297.5	2348.75
Median	1187.5	1186	2005	2072	2072	2005	2148	2214.5
Geometric Mean	1266.75	1268.56	2021.45	2030.50	2108.84	2021.45	2152.69	2215.67
Standard Deviation	1359.15	1358.24	1022.45	1034.25	971.87	1022.45	958.49	926.2568
Confidence	1331.94	1331.05	1001.98	1013.55	952.42	1001.98	939.31	907.71
Confidence Interval (+)	2953.19	2952.55	3193.23	3224.8	3208.67	3193.23	3236.81	3256.46
Confidence Interval (-)	289.30	290.44	1189.26	1197.7	1303.82	1189.26	1358.18	1441.03
Confidence Interval	2663.89	2662.1	2003.96	2027.1	1904.84	2003.96	1878.62	1815.43

System's Efficiency Evaluation In previous sub section, we evaluated the user-friendliness of the system by calculating the time taken by physicians on creating MLMs. Similarly, we calculated the number of errors in MLMs creation to evaluate the interoperability of the system. To convert these qualitative evaluation into quantitative, we evaluated the systems' efficiency. The system's efficiency depends on two important metrics *Time On Task*, and *Task Success Rate*.

Time On Task: It is also called task completion time or shortly task time, which is good metrics for measuring the efficiency of the system [119]. *Time On Task* is dependent on statistical values such as average, median, geometric mean, and confidence intervals. All these statistical values are contributing in the measurement of *Time On Task*. The statistical measurements of proposed system I-KAT are shown in Table 6.10, and the statistical measurements of ArdenSuite are shown Table 6.11 . The most common visualization of the *Time On Task* is to look at the

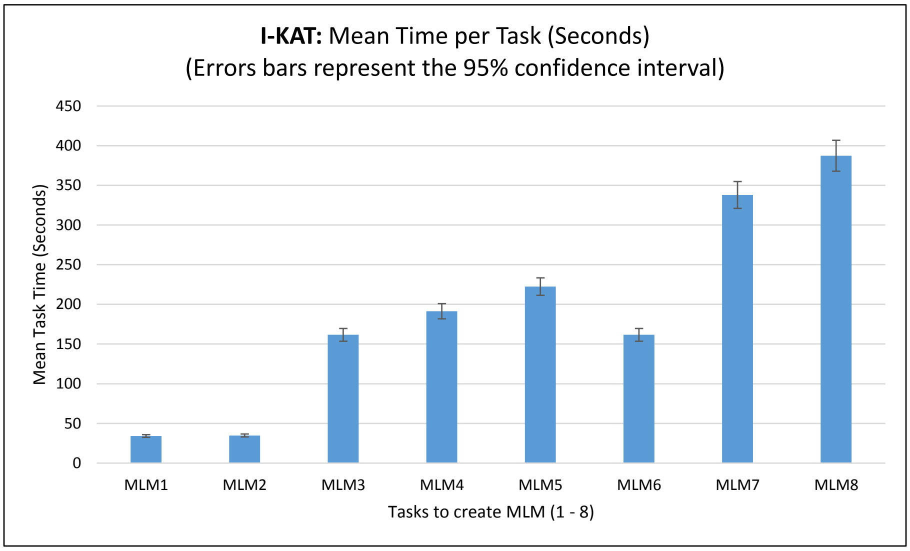

Figure 6.11: I-KAT: Mean time on task, in seconds. Error bars represent a 95% confidence interval.

mean values of the time spent on particular tasks, however, sometimes the median values are more appropriate to visualize and summarize the *Time On Task*. The most straightforward and intuitive way to visualize it through the mean values. When several users have taken exceedingly long time for a task completion, then it will be cause to increase the average. If there are several outliers then the values of confidence interval will be more appropriate to visualize the Time On Task. Therefore, we visualized the Time On Task for proposed system and existing system with respect to mean value, which are shown in Figure 6.11 and 6.12, respectively.

The error bars represent 95% confidence interval. The results show that out proposed system takes lesser time than the existing system in all simple and complex MLMs creation. Similarly, the error bars of our proposed system illustrates that our confidence interval is also lesser than existing system, which is positive observation. The less confidence interval describes that the experiment on creating new MLM will be taken approximately equal time to current *Time On Task*.

Task Success: The *Task Success Rate* is the most common usability metric in the area of user experience [119]. In Task Success evaluation, the task successful completion is measured in binary success. Each task has some defined and concrete goal, if it is achieved successfully then it is considered as pass in binary success, otherwise it is considered as fail. We evaluated our

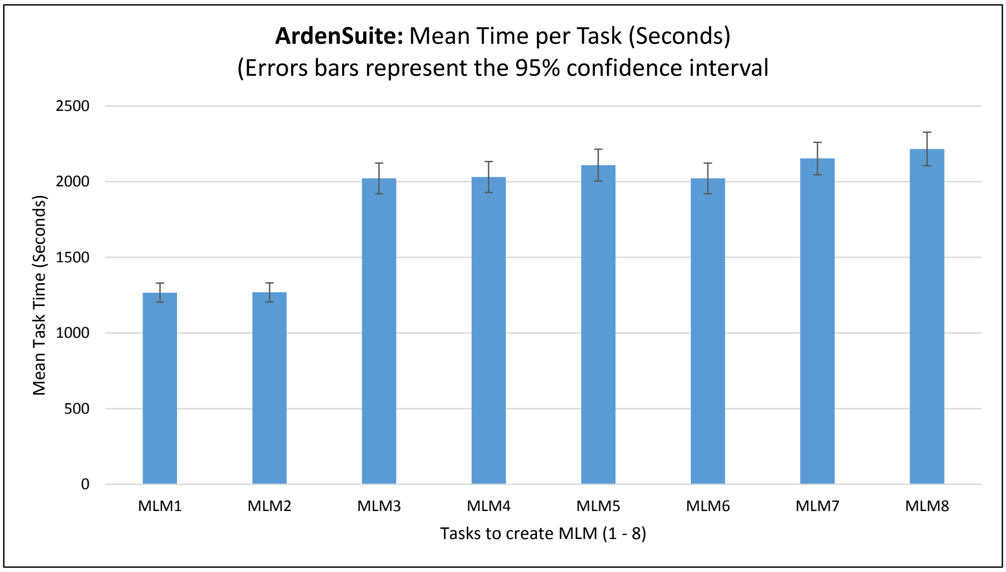

Figure 6.12: I-KAT: Mean time on task, in seconds. Error bars represent a 95% confidence interval.

proposed system with existing system based on the syntax and logical errors, but the MLM with syntax error cannot compile and share with other medical institutions. However, the logical errors give wrong results but the MLM is created successfully without syntax error. Our proposed system hides the structure and syntax of MLM from the physicians, therefore, our proposed system is error-free with respect to syntax error. Additionally, we also provided the controlled environment to physicians for easy selection of concepts during rule creation, it also decreases the chance of logical error but it is difficult to make the system error-free with respect to logical errors.

After experiments observation, we defined the success and failure criteria with threshold value of 2. The MLM created with more than 2 errors are considered as task failure and lesser than or equal to 2 then we considered as task success. The task successes are represented with 1's and task failures are represented with 0's. Table 6.12 shows the task success rate for our proposed system, which is 90.625%, and Table 6.13 shows the task success rate of the existing system, which is 46.875%.

Table 6.12: I-KAT: Task Success Rate for 4 participants and 8 MLMs.

Participants	MLM1	MLM2	MLM3	MLM4	MLM5	MLM6	MLM7	MLM8	Average
Physician 1	1	1	1	1	1	1	1	1	**100%**
Physician 2	1	1	1	1	1	1	1	1	**100%**
Physician 3	1	1	1	1	0	1	0	0	**62.5%**
Knowledge Engineer	1	1	1	1	1	1	1	1	**100%**
Average	**100%**	**100%**	**100%**	**100%**	**75%**	**100%**	**75%**	**75%**	**90.62%**

Table 6.13: ArdenSuite: Task Success Rate for 4 participants and 8 MLMs.

Participants	MLM1	MLM2	MLM3	MLM4	MLM5	MLM6	MLM7	MLM8	Average
Physician 1	1	1	1	0	0	0	0	0	**35.5%**
Physician 2	1	1	0	0	0	0	0	0	**25%**
Physician 3	1	1	0	0	0	0	0	0	**25%**
Knowledge Engineer	1	1	1	1	1	1	1	1	**100%**
Average	**100%**	**100%**	**50%**	**25%**	**25%**	**25%**	**25%**	**25%**	**46.87%**

The graphical representation of our proposed system (Figure 6.13) and existing system (Figure 6.14), showed that task success rate of proposed system is higher than existing system in all simplest and complex MLMs.

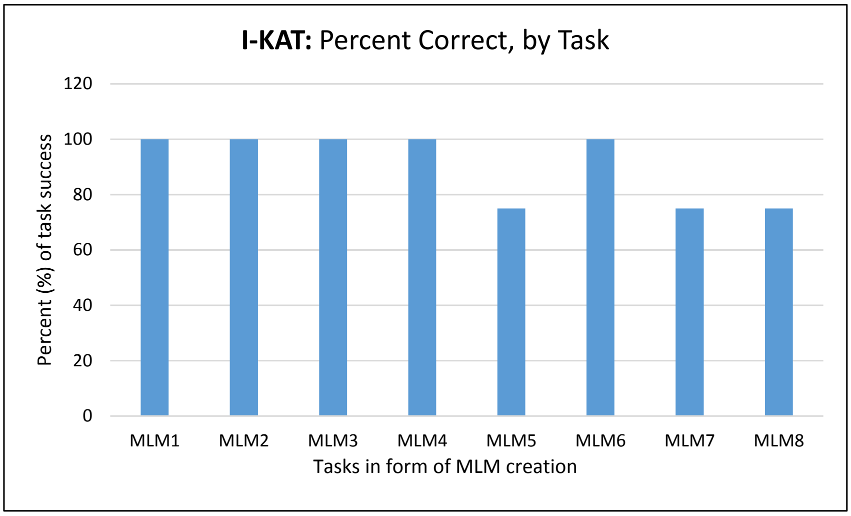

Figure 6.13: I-KAT: Task Success Rate in Percent.

Efficiency: The *System Efficiency* is usually measured by combination of *Time On Task* and *Task Success Rate* [119]. Common Industry Format for Usability Test Reports (ISO/IEC 25062:2006) defines the *System Efficiency* as a ratio of the *Task Success Rate* to the mean of *Time*

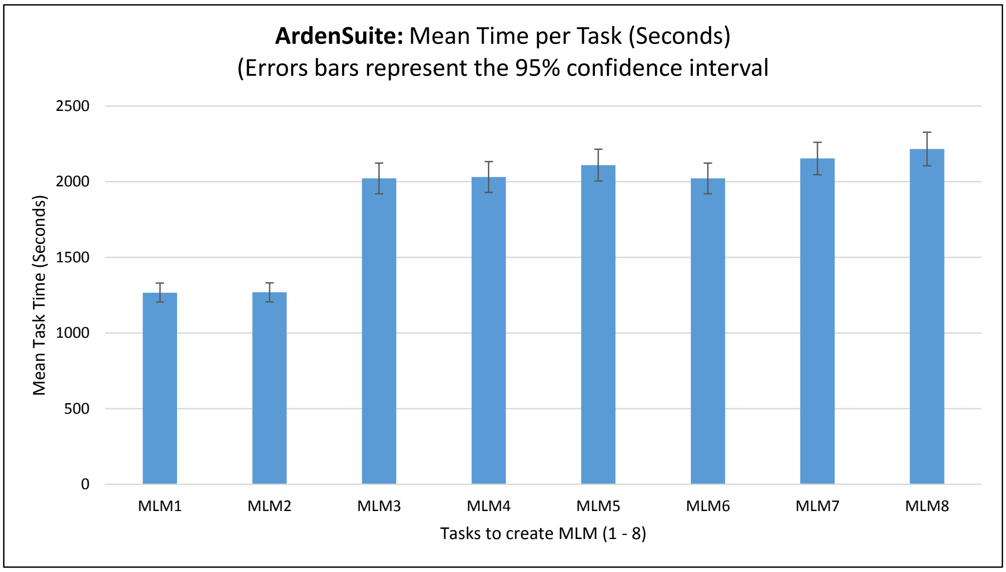

Figure 6.14: ArdenSuite: Task Success Rate in Percent.

On Task. In efficiency evaluation, we used the calculated *Task Success Rate* and *Time On Task* to find the system's efficiency. The efficiency of our proposed system is 56.625 completion rate/time, while the efficiency of the proposed system is 1.875 completion rate/time. Table 6.14 shows the efficiency of the proposed system for each MLM with respect to task success per minute, while Table 6.15 shows the efficiency of the existing system for each MLM.

Table 6.14: I-KAT: Efficiency Measurement.

Tasks	Task Success Rate	Time On Task (Sec)	Time On Task (min)	Efficiency
MLM1	100	38.5	0.64	**155**
MLM2	100	38.75	0.64	**154**
MLM3	100	162.5	2.70	**36**
MLM4	100	195	3.25	**30**
MLM5	75	225.5	3.75	**19**
MLM6	100	162.5	2.7	**36**
MLM7	75	350.25	5.83	**12**
MLM8	75	400.5	6.67	**11**
Overall Efficiency (I-KAT)				**56.62**

Table 6.15: ArdenSuite: Efficiency Measurement.

Tasks	Task Success Rate	Time On Task (Sec)	Time On Task (min)	Efficiency
MLM1	100	1621.25	27.02	**4**
MLM2	100	1621.5	27.02	**4**
MLM3	50	2191.25	36.52	**2**
MLM4	25	2211.25	36.85	**1**
MLM5	25	2256.25	37.6	**1**
MLM6	25	2191.25	36.52	**1**
MLM7	25	2297.5	38.29	**1**
MLM8	25	2348.75	39.14	**1**
Overall Efficiency (ArdenSuite)				**1.87**

Figure 6.15 illustrated the graphical representation of the proposed system's efficiency. It shows that the system's efficiency is very high in simple MLM, and gradually decreases according to the complexity of MLMs. The overall efficiency of the proposed system is much better than the existing system in each MLM, Figure 6.16 shows the efficiency of the existing system.

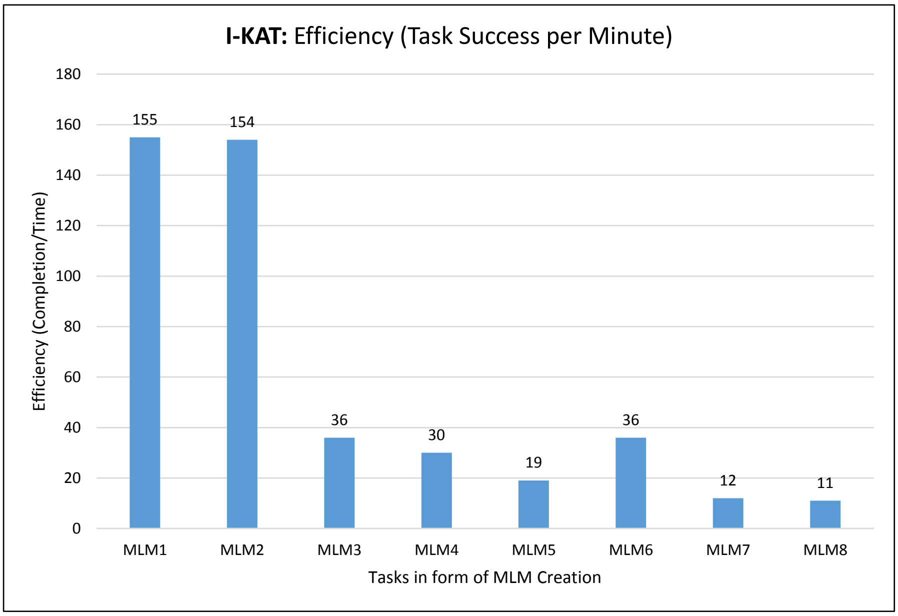

Figure 6.15: I-KAT: Efficiency Measurement.

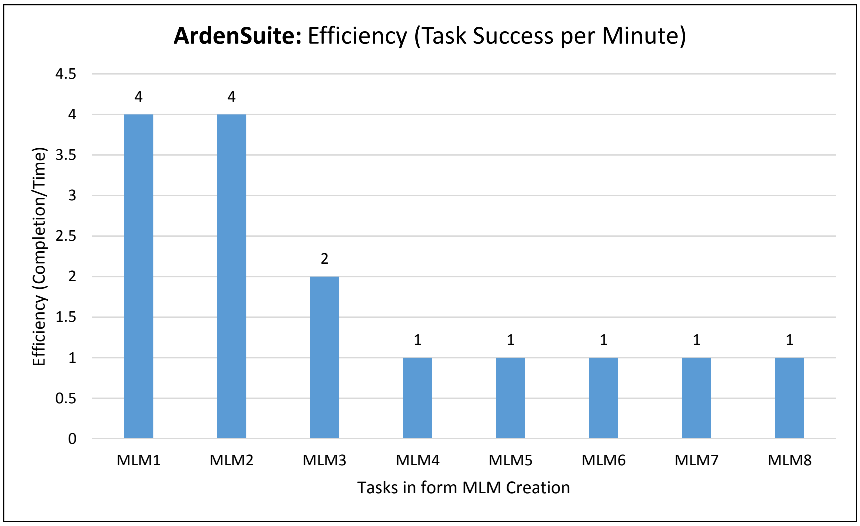

Figure 6.16: ArdenSuite: Efficiency Measurement.

6.3 Summary

Our proposed algorithm for *DCM-Standard Terminology Mapping* is evaluated with state-of-the-art systems with statistical measures such as precision, recall, and F-Measure. Our algorithm shows better results than the existing systems as precision 0.95, recall 0.92, and F-Measure 0.93. Usually, the ontology matching algorithms lack definition based algorithm, therefore, we evaluated our proposed definition based algorithm with base-line (Jaccard Similarity) algorithm for *Standard Terminology and Data Model Mapping*. As compared to Jaccard Similarity algorithm, our proposed algorithm showed better results as precision (0.89), recall (0.97), and F-measure (0.93).

For system centric evaluation, we compared implementation of clinical information modelling systems requirement in our proposed system and in existing system. The results suggested that 82.05% of the requirements were fully supported, 7.69% were partially supported, and 10.25% were not supported at all by our system. Whereas, in the existing systems, 35.89% are fully supported, 28.20% are partially supported, and 35.89% are not supported at all. In user centric evaluation the assessment criterion was ‘ease of use’. The proposed system showed 15 times better results with respect to time in MLM creation as compare to the existing systems. Moreover, the participants on average made only one error in MLM creation using our proposed system, while the average rate of error using existing systems was 13 per MLM. We also evaluated the proposed system efficiency with respect to time on task and task success rate. The task success rate of the proposed system was 90.625%, while the existing system was 46.87%. Based on the ratio of mean time completion to the success rate, the overall efficiency of the proposed system was 56.625, which better than the existing system efficiency 1.857 completion rate/time.

Chapter 7

Conclusion and Future Directions

7.1 Conclusion

Technologically integrated healthcare systems can be realized if physicians are encouraged to use smart systems in different phases of patient care such as diagnosis, treatment, and follow-ups. CDSS plays an important role in decision making. However, adaption of CDSS in clinical workflows is challenging due to the knowledge base evolution, according to continuous innovative research in medical domain. Existing systems lack interoperability and shareability of knowledge due to lack of practicing medical standards. The utilization of medical standards increases the knowledge creation complexity and overburdens the physicians to evolve the knowledge. Therefore, we proposed a *Semantic Reconciliation Model (SRM)* to create shareable and interoperable knowledge using a user-friendly authoring environment. Firstly, the proposed model provides *schema-data level semantic reconciliation* using flexible mapping methodology to achieve the knowledge interoperability goal. Secondly, the SRM provides *structure level semantic reconciliation* to create shareable knowledge. The convergence of medical standards can make the knowledge shareable and interoperable, and there exists many standards in the medical domain.

Arden Syntax is close to natural language, making it easier for physicians to understand and utilize it for knowledge rule creation. However, a number of complex artifacts in the Arden Syntax specification increase its complexity. Therefore, our proposed system provides simplified interfaces to hide the Arden Syntax complexity to some extent. Moreover, our system automatically generates MLMs using the maximum number of Arden Syntax artifacts. These artifacts include ":= object," ":= read," "EXTRACT ATTRIBUTE NAME," "IF THEN," and others, as shown in MLM Listing A.1 in Appendix A.2. However, some artifacts are not supported by our system, such as loops and some aggregate functions.

The existing legacy HMIS have diverse format of schemas to represent the system's internal data models. This diversity reduces data interoperability and increases the complexity for integrating CDSS with legacy HMIS systems. The HL7 community recommended the vMR standard data model as an appropriate solution. Existing systems define the input parameters of an MLM using curly braces to represent a query from an external system database, but the designed data models in databases are different. Therefore, the use of a standard data model, HL7 vMR, helps to remove the curly brace problem during integration of CDSS with legacy HMIS. The proposed system provides direction towards the objective of automatic compilation of Arden Syntax to executable format. Arden2ByteCode [73] and ArdenSuite [44] systems incorporate automatic compilation of Arden Syntax to executable format. Arden2ByteCode require physician's expertise in Eclipse framework, while ArdenSuite is a commercial product. Physician feels burden in understanding Eclipse environment, therefore, we intend as our future work, development of automatic compilation of Arden Syntax to executable format with fully integrated Arden Syntax MLM creation and testing environment. In the comparison evaluation, we evaluated our system with ArdenSuite, which is a commercially available system with mature compilation functionality. However, in the comparison, we only focused on the creation of shareable and interoperable knowledge in the form of MLMs.

SRM provides a flexible concept modelling environment to accommodate new concepts that can easily evolve using SOAP representation of DCM and data model vMR. We designed the DCM based on the well-known SOAP protocol, which provides a structured system for a comprehensive analysis of problems, diagnosis, treatment plans, demographics, and patient history [103]. Therefore, the DCM can easily adjust new concepts under one of its categories. Similarly, the data model vMR is envisioned to model CDSS-related clinical concepts and attributes with high scalability [107]. The data model vMR is designed and developed as a comprehensive and scalable representative set of data elements after a rigorous multi-national and multi-institutional analysis of CDSS systems [26].

We created *Domain Ontology* to provide a related value set in an IntelliSense window for user selection of the desired concept. Searching the related value set in the entire SNOMED CT versus in the *Domain Ontology* represents a tradeoff between performance efficiency and concept coverage. Searching the entire SNOMED CT for a concept improves coverage but slows performance at the interface level. Likewise, searching only the *Domain Ontology* decreases the concept coverage, but increases efficiency.

The current developed system validates new MLMs by comparing title, name, and purpose with previously created MLMs to find duplicates. We are conducting ongoing research for the maintenance and validation of MLMs; in the future, the system will examine the logic of new rules to determine whether these already exist in the MLM repository. Our experiments show that even novice users were able to create MLMs using our system compared to ArdenSuite. This demonstrates that our system provides a very user-friendly environment that enables physicians with minimal Arden Syntax experience to share their knowledge.

7.2 Future Directions

According to multi-model mapping, we will evaluate our system on large biomedical ontologies and compare with existing systems. We will participate in Ontology Alignment Evaluation Initiative (OAEI) competition in near future. We also aim to extend the system with further mappings between vMR and DCM concepts to support a higher number of concepts.

Additionally, we endeavor to integrate our ongoing research on maintenance and validation of MLMs into the current system. We plan to extend the system to support complex Arden Syntax artifacts such as loops and aggregate functions. Similarly, the state-of-the-art CDSS Hook is currently popular as a standard, therefore, we will also implement the system for CDSS Hook.

Bibliography

[1] M. Hussain, A. Khattak, W. Khan, I. Fatima, M. Amin, Z. Pervez, R. Batool, M. Saleem, M. Afzal, M. Faheem *et al.*, "Cloud-based smart cdss for chronic diseases," *Health and Technology*, vol. 3, no. 2, pp. 153–175, 2013.

[2] M. Hussain, M. Afzal, T. Ali, R. Ali, W. A. Khan, A. Jamshed, S. Lee, B. H. Kang, and K. Latif, "Data-driven knowledge acquisition, validation, and transformation into hl7 arden syntax," *Artificial intelligence in medicine*, 2015.

[3] E. Sanchez, C. Toro, E. Carrasco, G. Bueno, C. Parra, P. Bonachela, M. Graña, and F. Guijarro, "An architecture for the semantic enhancement of clinical decision support systems," in *Knowlege-Based and Intelligent Information and Engineering Systems*. Springer, 2011, pp. 611–620.

[4] A. Wright and D. F. Sittig, "A framework and model for evaluating clinical decision support architectures," *Journal of biomedical informatics*, vol. 41, no. 6, pp. 982–990, 2008.

[5] S. Bakken, H. Jia, E. S. Chen, J. Choi, R. M. John, N.-J. Lee, E. Mendonca, W. D. Roberts, O. Velez, and L. M. Currie, "The effect of a mobile health decision support system on diagnosis and management of obesity, tobacco use, and depression in adults and children," *The Journal for Nurse Practitioners*, vol. 10, no. 10, pp. 774–780, 2014.

[6] A. X. Garg, N. K. Adhikari, H. McDonald, M. P. Rosas-Arellano, P. Devereaux, J. Beyene, J. Sam, and R. B. Haynes, "Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review," *Jama*, vol. 293, no. 10, pp. 1223–1238, 2005.

[7] "2015 edition health information technology (health it) certification criteria, 2015 edition base electronic health record (ehr) definition, and onc health it certification program modifications," https://www.federalregister.gov/documents/2015/03/30/2015-06612/2015-edition-health-information-technology-health-it-certification-criteria-2015-edition-base, 2017, (Last visited in March 2017).

[8] M. Hussain, M. Afzal, W. A. Khan, and S. Lee, "Clinical decision support service for elderly people in smart home environment," in *Control Automation Robotics & Vision (ICARCV), 2012 12th International Conference on*. IEEE, 2012, pp. 678–683.

[9] C. Y. Jung, K. A. Sward, and P. J. Haug, "Executing medical logic modules expressed in ardenml using drools," *Journal of the American Medical Informatics Association*, vol. 19, no. 4, pp. 533–536, 2012.

[10] Y. Zhang, S. Wang, P. Sun, and P. Phillips, "Pathological brain detection based on wavelet entropy and hu moment invariants," *Bio-medical materials and engineering*, vol. 26, no. s1, pp. S1283–S1290, 2015.

[11] Y.-D. Zhang, S. Chen, S.-H. Wang, J.-F. Yang, and P. Phillips, "Magnetic resonance brain image classification based on weighted-type fractional fourier transform and nonparallel support vector machine," *International Journal of Imaging Systems and Technology*, vol. 25, no. 4, pp. 317–327, 2015.

[12] J. Moss and E. S. Berner, "Evaluating clinical decision support tools for medication administration safety in a simulated environment," *International journal of medical informatics*, vol. 84, no. 5, pp. 308–318, 2015.

[13] K. Fehre and K.-P. Adlassnig, "Service-oriented arden-syntax-based clinical decision support," *Proceedings of eHealth2011. Vienna: Austrian Computer Society*, pp. 123–8, 2011.

[14] J. A. Osheroff, J. M. Teich, B. Middleton, E. B. Steen, A. Wright, and D. E. Detmer, "A roadmap for national action on clinical decision support," *Journal of the American medical informatics association*, vol. 14, no. 2, pp. 141–145, 2007.

[15] A. Wright, D. F. Sittig, J. S. Ash, S. Sharma, J. E. Pang, and B. Middleton, "Clinical decision support capabilities of commercially-available clinical information systems," *Journal of the American Medical Informatics Association*, vol. 16, no. 5, pp. 637–644, 2009.

[16] H. Varonen, T. Kortteisto, M. Kaila, and E. S. Group, "What may help or hinder the implementation of computerized decision support systems (cdsss): a focus group study with physicians," *Family practice*, vol. 25, no. 3, pp. 162–167, 2008.

[17] A. Wright and D. F. Sittig, "A four-phase model of the evolution of clinical decision support architectures," *International journal of medical informatics*, vol. 77, no. 10, pp. 641–649, 2008.

[18] T. Ali, M. Hussain, W. A. Khan, M. Afzal, B. H. Kang, and S. Lee, "Arden syntax studio: Creating medical logic module as shareable knowledge," in *Innovations in Intelligent Systems and Applications (INISTA) Proceedings, 2014 IEEE International Symposium on*. IEEE, 2014, pp. 266–272.

[19] D. F. Sittig, A. Wright, J. A. Osheroff, B. Middleton, J. M. Teich, J. S. Ash, E. Campbell, and D. W. Bates, "Grand challenges in clinical decision support," *Journal of biomedical informatics*, vol. 41, no. 2, pp. 387–392, 2008.

[20] J. Choi, Y. A. Lussier, and E. A. Mendonça, "Adapting current arden syntax knowledge for an object oriented event monitor," in *AMIA Annual Symposium Proceedings*, vol. 2003. American Medical Informatics Association, 2003, p. 814.

[21] H. K. Park, J. pil Kim, J. W. Choi, J. J. Hwang, and S. M. Ha, "Design of real-time arden syntax based decision support system with minimum workload and building cost," in *Enterprise networking and Computing in Healthcare Industry, 2005. HEALTHCOM 2005. Proceedings of 7th International Workshop on*. IEEE, 2005, pp. 430–433.

[22] G. Hripcsak, P. Ludemann, T. A. Pryor, O. B. Wigertz, and P. D. Clayton, "Rationale for the arden syntax," *Computers and Biomedical Research*, vol. 27, no. 4, pp. 291–324, 1994.

[23] R. Jenders, H. Huang, G. Hripcsak, and P. Clayton, “Evolution of a knowledge base for a clinical decision support system encoded in the arden syntax.” in *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 1998, p. 558.

[24] A. Moreno-Conde, F. Jódar-Sánchez, and D. Kalra, “Requirements for clinical information modelling tools,” *International journal of medical informatics*, vol. 84, no. 7, pp. 524–536, 2015.

[25] W. A. Khan, A. M. Khattak, M. Hussain, M. B. Amin, M. Afzal, C. Nugent, and S. Lee, “An adaptive semantic based mediation system for data interoperability among health information systems,” *Journal of medical systems*, vol. 38, no. 8, pp. 1–18, 2014.

[26] K. Kawamoto, G. Del Fiol, H. R. Strasberg, N. Hulse, C. Curtis, J. J. Cimino, B. H. Rocha, S. Maviglia, E. Fry, H. J. Scherpbier *et al.*, “Multi-national, multi-institutional analysis of clinical decision support data needs to inform development of the hl7 virtual medical record standard,” in *AMIA Annual Symposium Proceedings*, vol. 2010. American Medical Informatics Association, 2010, p. 377.

[27] S. Kim, P. J. Haug, R. A. Rocha, and I. Choi, “Modeling the arden syntax for medical decisions in xml,” *International journal of medical informatics*, vol. 77, no. 10, pp. 650–656, 2008.

[28] P. D. Johnson, S. W. Tu, M. Musen, and I. Purves, “A virtual medical record for guideline-based decision support.” in *Proceedings of the AMIA symposium*. American Medical Informatics Association, 2001, p. 294.

[29] “Welcome to snomed international,” https://www.snomed.org/, 2018, (Last visited in August 2018).

[30] S. M. Huff, R. A. Rocha, C. J. McDonald, G. J. De Moor, T. Fiers, W. D. Bidgood, A. W. Forrey, W. G. Francis, W. R. Tracy, D. Leavelle *et al.*, “Development of the logical observation identifier names and codes (loinc) vocabulary,” *Journal of the American Medical Informatics Association*, vol. 5, no. 3, pp. 276–292, 1998.

[31] Z. Aleksovski and M. Sevenster, “Identifying breast cancer concepts in SNOMED-CT using large text corpus,” in *Electronic Healthcare*. Springer, 2012, pp. 27–34.

[32] I. Ben Hamouda, M. Feki, I. Boughzala, and O. Chourabi, “Understanding knowledge sharing in health care system,” 2015.

[33] O. CHOURABI *et al.*, “Understanding the barriers to knowledge sharing in the french healthcare system: a conceptual model and research propositions,” 2018.

[34] R. Rezaei, T. K. Chiew, S. P. Lee, and Z. S. Aliee, “Interoperability evaluation models: A systematic review,” *Computers in Industry*, vol. 65, no. 1, pp. 1–23, 2014.

[35] A. Gyrard, C. Bonnet, and K. Boudaoud, “Domain knowledge interoperability to build the semantic web of things,” in *W3C Workshop on the Web of Things*, 2014, pp. 1–5.

[36] R. Sottilare, A. Graesser, X. Hu, and K. Brawner, *Design recommendations for intelligent tutoring systems: authoring tools and expert modeling techniques*. Robert Sottilare, 2015.

[37] E. O’Donnell, S. Lawless, M. Sharp, and V. P. Wade, “A review of personalised e-learning: Towards supporting learner diversity,” *International Journal of Distance Education Technologies (IJDET)*, vol. 13, no. 1, pp. 22–47, 2015.

[38] R. A. Jenders and B. Dasgupta, “Challenges in implementing a knowledge editor for the arden syntax: knowledge base maintenance and standardization of database linkages.” in *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 2002, p. 355.

[39] T. Ali, M. Hussain, W. Ali Khan, M. Afzal, and S. Lee, “Authoring tool: Acquiring sharable knowledge for smart cdss,” in *Engineering in Medicine and Biology Society (EMBS), 2013 35th Annual International Conference of the IEEE*. IEEE, 2013, pp. 1278–1281.

[40] V. L. Patel, J. F. Arocha, and A. W. Kushniruk, “Patients’ and physicians’ understanding of health and biomedical concepts: relationship to the design of EMR systems,” *Journal of biomedical informatics*, vol. 35, no. 1, pp. 8–16, 2002.

[41] D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto, “The agreement-makerlight ontology matching system,” in *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*. Springer, 2013, pp. 527–541.

[42] L. Otero-Cerdeira, F. J. Rodríguez-Martínez, and A. Gómez-Rodríguez, “Definition of an ontology matching algorithm for context integration in smart cities,” *Sensors*, vol. 14, no. 12, pp. 23 581–23 619, 2014.

[43] S. Schulz and C. Martínez-Costa, “How ontologies can improve semantic interoperability in health care,” in *Process Support and Knowledge Representation in Health Care*. Springer, 2013, pp. 1–10.

[44] M. Samwald, K. Fehre, J. De Bruin, and K.-P. Adlassnig, “The arden syntax standard for clinical decision support: Experiences and directions,” *Journal of biomedical informatics*, vol. 45, no. 4, pp. 711–718, 2012.

[45] “ARDENSUITE, medical knowledge representation and rule-based inference software with arden syntax,” http://www.medexter.com/arden-syntax/arden-syntax, 2017, (Last visited in March 2017).

[46] M. Achichi, M. Cheatham, Z. Dragisic, J. Euzenat, D. Faria, A. Ferrara, G. Flouris, I. Fundulaki, I. Harrow, V. Ivanova *et al.*, “Results of the ontology alignment evaluation initiative 2017,” in *OM 2017-12th ISWC workshop on ontology matching*. No commercial editor., 2017, pp. 61–113.

[47] W. Hu and Y. Qu, “Falcon-ao: A practical ontology matching system,” *Web semantics: science, services and agents on the world wide web*, vol. 6, no. 3, pp. 237–239, 2008.

[48] P. Shvaiko and J. Euzenat, “Ontology matching: state of the art and future challenges,” *IEEE Transactions on knowledge and data engineering*, vol. 25, no. 1, pp. 158–176, 2013.

[49] B. Xu, P. Wang, J. Lu, Y. Li, and D. Kang, “Theory and semantic refinement of bridge ontology based on multi-ontologies,” in *Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on*. IEEE, 2004, pp. 442–449.

[50] F. Jauro, S. Junaidu, and S. Abdullahi, "Falcon-ao++: An improved ontology alignment system," *International Journal of Computer Applications*, vol. 94, no. 2, 2014.

[51] A. Groß, M. Hartung, T. Kirsten, and E. Rahm, "Gomma results for oaei 2012," in *Proceedings of the 7th International Conference on Ontology Matching-Volume 946*. CEUR-WS. org, 2012, pp. 133–140.

[52] F. Song, G. Zacharewicz, and D. Chen, "An ontology-driven framework towards building enterprise semantic information layer," *Advanced Engineering Informatics*, vol. 27, no. 1, pp. 38–50, 2013.

[53] I. F. Cruz, F. P. Antonelli, and C. Stroe, "Agreementmaker: efficient matching for large real-world schemas and ontologies," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1586–1589, 2009.

[54] D. Ngo, Z. Bellahsene, and R. Coletta, "A flexible system for ontology matching," in *Forum at the Conference on Advanced Information Systems Engineering (CAiSE)*. Springer, 2011, pp. 79–94.

[55] D. Ngo and Z. Bellahsene, "Overview of yam++—(not) yet another matcher for ontology alignment task," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 41, pp. 30–49, 2016.

[56] ——, "Yam++: a multi-strategy based approach for ontology matching task," in *International Conference on Knowledge Engineering and Knowledge Management*. Springer, 2012, pp. 421–425.

[57] E. Jiménez-Ruiz and B. C. Grau, "Logmap: Logic-based and scalable ontology matching," in *International Semantic Web Conference*. Springer, 2011, pp. 273–288.

[58] M. KACHROUDI, G. DIALLO, and S. B. YAHIA, "Oaei 2017 results of kepler," in *OM-2017: Proceedings of the Twelfth International Workshop on Ontology Matching*, 2017, p. 138.

[59] W. E. Djeddi and M. T. Khadir, “Xmap: a novel structural approach for alignment of owl-full ontologies,” in *Machine and Web Intelligence (ICMWI), 2010 International Conference on*. IEEE, 2010, pp. 368–373.

[60] X. Xue and J.-S. Pan, “A segment-based approach for large-scale ontology matching,” *Knowledge and Information Systems*, vol. 52, no. 2, pp. 467–484, 2017.

[61] A. Annane, Z. Bellahsene, F. Azouaou, and C. Jonquet, “Yam-bio–results for oaei 2017,” 2017.

[62] X. Chen, W. Xia, E. Jiménez-Ruiz, and V. V. Cross, “Extending an ontology alignment system with bioportal: a preliminary analysis.” in *International Semantic Web Conference (Posters & Demos)*. Citeseer, 2014, pp. 313–316.

[63] “Foundational model of anatomy,” https://bioportal.bioontology.org/ontologies/FMA, 2018, (Last visited in August 2018).

[64] “National cancer institute thesaurus,” https://bioportal.bioontology.org/ontologies/NCIT, 2018, (Last visited in August 2018).

[65] I. M. F. R. O. T. A. Laadhar, F. Ghozzi and F. Gargouri, “Omap: An effective pairwise ontology matching system,” *Proceedings of the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, vol. 2, no. 2, pp. 161–168, 2017.

[66] A. Laadhar, F. Ghozzi, I. Megdiche, F. Ravat, O. Teste, and F. Gargouri, “Pomap results for oaei 2017,” in *12th International Workshop on Ontology Matching collocated with the 16th International Semantic Web Conference (OM@ ISWC’17)*, 2017, pp. pp–1.

[67] M. Mohammadia, A. Atashinb, W. Hofmanc, and Y.-H. Tana, “Sanom results for oaei 2017,” in *OM-2017: Proceedings of the Twelfth International Workshop on Ontology Matching*, 2017, p. 185.

[68] S. Hertling and H. Paulheim, “Wikimatch: using wikipedia for ontology matching,” *Ontology Matching*, vol. 946, 2012.

[69] S. Hertling, "Wikiv3 results for oaei 2017," in *OM-2017: Proceedings of the Twelfth International Workshop on Ontology Matching*, 2017, p. 190.

[70] A. A. Boxwala, M. Peleg, S. Tu, O. Ogunyemi, Q. T. Zeng, D. Wang, V. L. Patel, R. A. Greenes, and E. H. Shortliffe, "Glif3: a representation format for sharable computer-interpretable clinical practice guidelines," *Journal of biomedical informatics*, vol. 37, no. 3, pp. 147–161, 2004.

[71] I. Cho, J. Kim, J. H. Kim, H. Y. Kim, and Y. Kim, "Design and implementation of a standards-based interoperable clinical decision support architecture in the context of the korean ehr," *International journal of medical informatics*, vol. 79, no. 9, pp. 611–622, 2010.

[72] R. A. Jenders, K.-P. Adlassnig, K. Fehre, and P. Haug, "Evolution of the arden syntax: Key technical issues from the standards development organization perspective," *Artificial Intelligence in Medicine*, 2016.

[73] M. Gietzelt, U. Goltz, D. Grunwald, M. Lochau, M. Marschollek, B. Song, and K.-H. Wolf, "Arden2bytecode: a one-pass arden syntax compiler for service-oriented decision support systems based on the osgi platform," *Computer methods and programs in biomedicine*, vol. 106, no. 2, pp. 114–125, 2012.

[74] V. Anand, A. E. Carroll, P. G. Biondich, T. M. Dugan, and S. M. Downs, "Pediatric decision support using adapted arden syntax," *Artificial intelligence in medicine*, 2015.

[75] S. L. Achour, M. Dojat, C. Rieux, P. Bierling, and E. Lepage, "A umls-based knowledge acquisition tool for rule-based clinical decision support system development," *Journal of the American Medical Informatics Association*, vol. 8, no. 4, pp. 351–360, 2001.

[76] D. Dunsmuir, J. Daniels, C. Brouse, S. Ford, and J. M. Ansermino, "A knowledge authoring tool for clinical decision support," *Journal of clinical monitoring and computing*, vol. 22, no. 3, pp. 189–198, 2008.

[77] A. Seitinger, K. Fehre, K.-P. Adlassnig, A. Rappelsberger, E. Wurm, E. Aberer, and M. Binder, "An arden-syntax-based clinical decision support framework for medical guide-

lines—lyme borreliosis as an example," *Studies in health technology and informatics*, vol. 198, pp. 125–132, 2013.

[78] N. C. Hulse, R. A. Rocha, G. Del Fiol, R. L. Bradshaw, T. P. Hanna, and L. K. Roemer, "Kat: a flexible xml-based knowledge authoring environment," *Journal of the American Medical Informatics Association*, vol. 12, no. 4, pp. 418–430, 2005.

[79] R. M. Sailors, "Mlm writer: An integrated development environment for creating arden syntax medical logic modules," in *Proceedings of the AMIA Annual Fall Symposium*. American Medical Informatics Association, 1996, p. 955.

[80] R. Jenders and B. Dasgupta, "Assessment of a knowledge-acquisition tool for writing medical logic modules in the arden syntax." in *Proceedings of the AMIA Annual Fall Symposium*. American Medical Informatics Association, 1996, p. 567.

[81] T. Ali, M. Hussain, W. A. Khan, M. Afzal, J. Hussain, R. Ali, W. Hassan, A. Jamshed, B. H. Kang, and S. Lee, "Multi-model-based interactive authoring environment for creating shareable medical knowledge," *Computer methods and programs in biomedicine*, vol. 150, pp. 41–72, 2017.

[82] W. A. Khan, M. B. Amin, A. M. Khattak, M. Hussain, M. Afzal, S. Lee, and E. S. Kim, "Object-oriented and ontology-alignment patterns-based expressive mediation bridge ontology (mbo)," *Journal of Information Science*, vol. 41, no. 3, pp. 296–314, 2015.

[83] W. A. Khan, M. Hussain, M. Afzal, M. B. Amin, M. A. Saleem, and S. Lee, "Personalized-detailed clinical model for data interoperability among clinical standards," *TELEMEDICINE and e-HEALTH*, vol. 19, no. 8, pp. 632–642, 2013.

[84] "Hl7 version 3 standard," http://www.hl7.org/implement/standards/product_brief.cfm, 2017, (Last visited in March 2017).

[85] M. Marcos, J. A. Maldonado, B. Martínez-Salvador, D. Boscá, and M. Robles, "Interoperability of clinical decision-support systems and electronic health records using archetypes: a case study in clinical trial eligibility," *Journal of biomedical informatics*, vol. 46, no. 4, pp. 676–689, 2013.

[86] W. A. Khan, M. Hussain, K. Latif, M. Afzal, F. Ahmad, and S. Lee, “Process interoperability in healthcare systems with dynamic semantic web services,” *Computing*, vol. 95, no. 9, pp. 837–862, 2013.

[87] W. R. A.-M. S. M. W. C. M. G. W. Beeler, S. Huff and G. Schadow, “Message development framework,” *Version 3.3, Health Level Seven, Inc., Tech. Rep*, December 1999.

[88] K. Donnelly, “Snomed-ct: The advanced terminology and coding system for ehealth,” *Studies in health technology and informatics*, vol. 121, p. 279, 2006.

[89] M. Q. Stearns, C. Price, K. A. Spackman, and A. Y. Wang, “Snomed clinical terms: overview of the development process and project status.” in *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 2001, p. 662.

[90] “Ihtsdo international health terminology standards development organization,” https://www.iso.org/organization/567570.html, 2017, (Last visited in October 2017).

[91] E. S. Berner, *Clinical decision support systems*. Springer, 2007, vol. 233.

[92] P. A. De Clercq, J. A. Blom, H. H. Korsten, and A. Hasman, “Approaches for creating computer-interpretable guidelines that facilitate decision support,” *Artificial intelligence in medicine*, vol. 31, no. 1, pp. 1–27, 2004.

[93] L. Ohno-Machado, J. H. Gennari, S. N. Murphy, N. L. Jain, S. W. Tu, D. E. Oliver, E. Pattison-Gordon, R. A. Greenes, E. H. Shortliffe, and G. O. Barnett, “The guideline interchange format: a model for representing guidelines,” *Journal of the American Medical Informatics Association*, vol. 5, no. 4, pp. 357–372, 1998.

[94] L. Raymond, G. Paré, D. Guinea, A. Ortiz, P. Poba-Nzaou, M.-C. Trudel, J. Marsan, and T. Micheneau, “The contribution of office-based EMR systems to the performance of family physicians and primary care medical practices,” in *System Sciences (HICSS), 2015 48th Hawaii International Conference on*. IEEE, 2015, pp. 3033–3042.

[95] G. Paré, L. Raymond, A. O. de Guinea, P. Poba-Nzaou, M.-C. Trudel, J. Marsan, and T. Micheneau, “Barriers to organizational adoption of EMR systems in family physician

practices: a mixed-methods study in canada," *International journal of medical informatics*, vol. 83, no. 8, pp. 548–558, 2014.

[96] M. Hussain, W. A. Khan, M. Afzal, and S. Lee, "Smart cdss for smart homes," in *International Conference on Smart Homes and Health Telematics*. Springer, 2012, pp. 266–269.

[97] R. Qamar, A. Rector *et al.*, "Semantic issues in integrating data from different models to achieve data interoperability," in *Medinfo 2007: Proceedings of the 12th World Congress on Health (Medical) Informatics; Building Sustainable Health Systems*. IOS Press, 2007, p. 674.

[98] "Object management group business process model and notation," http://www.bpmn.org/, 2017, (Last visited in February 2017).

[99] S. Ferrante, S. Bonacina, G. Pozzi, F. Pinciroli, S. Marceglia *et al.*, "A design methodology for medical processes," *Applied clinical informatics*, vol. 7, no. 1, pp. 191–210, 2016.

[100] "Exceptional high-end modeling power," http://www.sparxsystems.com/products/ea/, 2017, (Last visited in February 2017).

[101] J. E. Frenzel, "Using electronic medical records to teach patient-centered care," *American journal of pharmaceutical education*, vol. 74, no. 4, 2010.

[102] L. L. Weed, "Medical records that guide and teach," *New England Journal of Medicine*, vol. 278, no. 12, pp. 652–657, 1968.

[103] J. Kibble, P. A. Hansen, and L. Nelson, "Use of modified soap notes and peer-led small-group discussion in a medical physiology course: addressing the hidden curriculum," *Advances in physiology education*, vol. 30, no. 4, pp. 230–236, 2006.

[104] S. Cameron and I. Turtle-Song, "Learning to write case notes using the soap format," *Journal of Counseling and Development: JCD*, vol. 80, no. 3, p. 286, 2002.

[105] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified modeling language reference manual, the*. Pearson Higher Education, 2004.

[106] P. J. Frederiks and T. P. Van der Weide, "Information modeling: The process and the required competencies of its participants," *Data & Knowledge Engineering*, vol. 58, no. 1, pp. 4–20, 2006.

[107] "HL7 Version 3 Standard," http://www.hl7.org, 2017, (Last visited in March 2017).

[108] "SNOMED CT Starter Guide," http://ihtsdo.org, 2017, (Last visited in March 2017).

[109] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[110] H. Liu and P. Singh, "Conceptnet—a practical commonsense reasoning tool-kit," *BT technology journal*, vol. 22, no. 4, pp. 211–226, 2004.

[111] "Allacronyms 3,134,000 acronyms and abbreviations. 16,440 topics," https://www.allacronyms.com, 2018, (Last visited in August 2018).

[112] V. Christen, A. Groß, J. Varghese, M. Dugas, and E. Rahm, "Annotating medical forms using umls," in *International Conference on Data Integration in the Life Sciences*. Springer, 2015, pp. 55–69.

[113] M. T. Pilehvar and R. Navigli, "From senses to texts: An all-in-one graph-based approach for measuring semantic similarity," *Artificial Intelligence*, vol. 228, pp. 95–128, 2015.

[114] S. Banerjee and T. Pedersen, "Extended gloss overlaps as a measure of semantic relatedness," in *Ijcai*, vol. 3, 2003, pp. 805–810.

[115] F. Baader, R. Küsters, and R. Molitor, "Computing least common subsumers in description logics with existential restrictions," in *IJCAI*, vol. 99, 1999, pp. 96–101.

[116] "National Comprehensive Cancer Network," http://www.nccn.org, 2017, (Last visited in February 2017).

[117] P. Cremonesi, F. Garzotto, and R. Turrin, "User-centric vs. system-centric evaluation of recommender systems," in *Human-Computer Interaction–INTERACT 2013*. Springer, 2013, pp. 334–351.

[118] R. Ali, M. Afzal, M. Hussain, M. Ali, M. H. Siddiqi, S. Lee, and B. H. Kang, "Multimodal hybrid reasoning methodology for personalized wellbeing services," *Computers in Biology and Medicine*, 2015.

[119] W. Albert and T. Tullis, *Measuring the user experience: collecting, analyzing, and presenting usability metrics*. Newnes, 2013.

[120] M. Afzal, M. Hussain, W. A. Khan, T. Ali, S. Lee, and B. H. Kang, "Knowledgebutton: An evidence adaptive tool for cdss and clinical research," in *Innovations in Intelligent Systems and Applications (INISTA) Proceedings, 2014 IEEE International Symposium on*. IEEE, 2014, pp. 273–280.

Appendix A

Implementation Details

A.1 System Implementation and Realization

We developed a web-based system and deployed it in a testing environment. In the user interface, we provided different screens such as *Rules List*, *Rule Editor*, and *Rule Viewer*[1]. The system provides a list of previously created MLMs with abstract information about the MLM, as shown in Figure A.1. This interface allows a physician to view and update the complete details of previously created MLMs. When a physician clicks on the *View* button, the corresponding MLM is shown in the *Rule Editor* in an editable form. The list screen also provides the functionality of adding a new rule through the *Create New Rule* button.

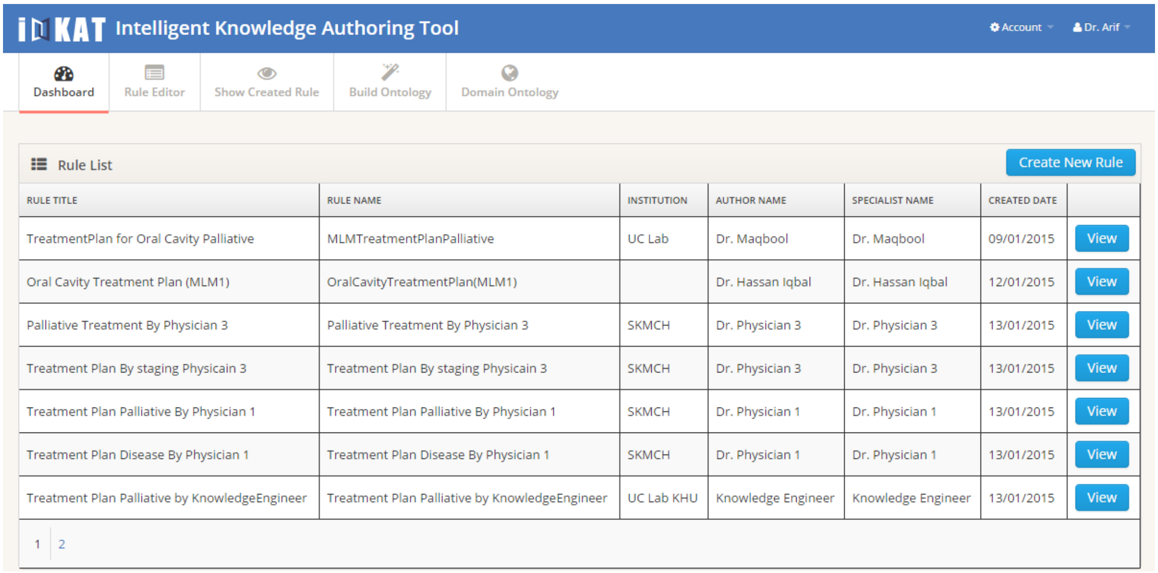

Figure A.1: Dashboard for existing MLM.

[1] Video Demo for review process: Please download video of rule creation using I-KAT https://goo.gl/Y8eHeu

The main interface for rule creation is *Rule Editor*, as shown in Figure A.2. Highlighted Area 1 is used to capture metadata about the MLM such as *Rule Title*, *Author's name*, *MLM name*, *Institution*, and *Created date*. Similarly, the physician can use the *Citation* button to attach online resources as evidence of the MLM [120]. The *Purpose* and *Explanation* boxes allow the physician to enter the rule purpose and provide explanation.

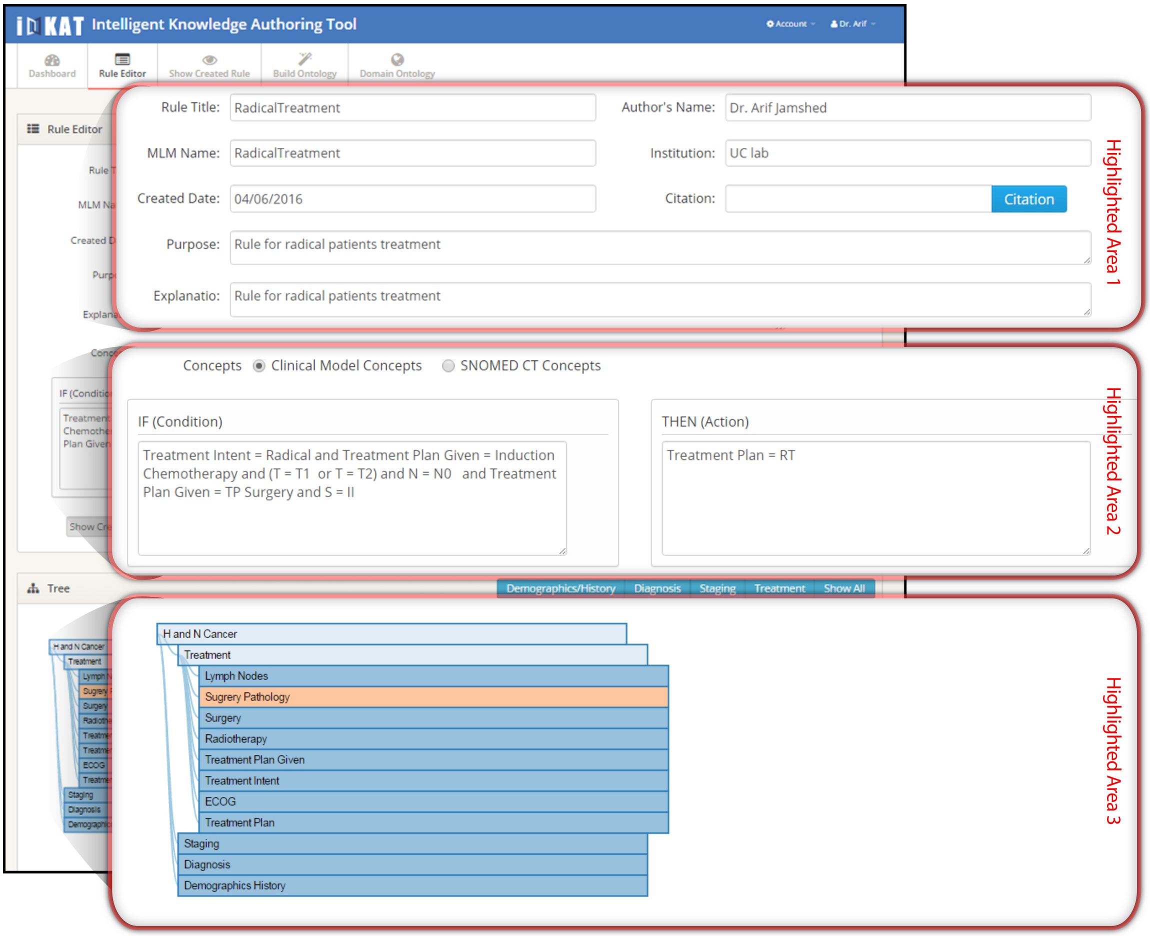

Figure A.2: Rule editor for MLM creation.

Highlighted Area 2 handles the main logic of the rule. It contains two boxes: *Condition* and *Action*. The *Condition* box allows the physician to write the facts involved in the condition part of the rule. The *Action* box is used to write the conclusion of the rule. This interface alleviates the physician from knowing the technical details of SNOMED CT, HL7 vMR, and the complex artifacts of HL7 Arden Syntax.

Physicians can select the DCM concepts from the DCM concepts tree shown in highlighted Area 3 by double-clicking on the required concept. The *Domain Clinical Model Concepts* option allows the domain concept to be brought from DCM, and *SNOMED CT Concepts* provides an enhanced search on the SNOMED ontology to obtain the domain concepts. While writing a condition or action statement, the physician can use either the tree model or the IntelliSense feature, as shown in Figure A.3.

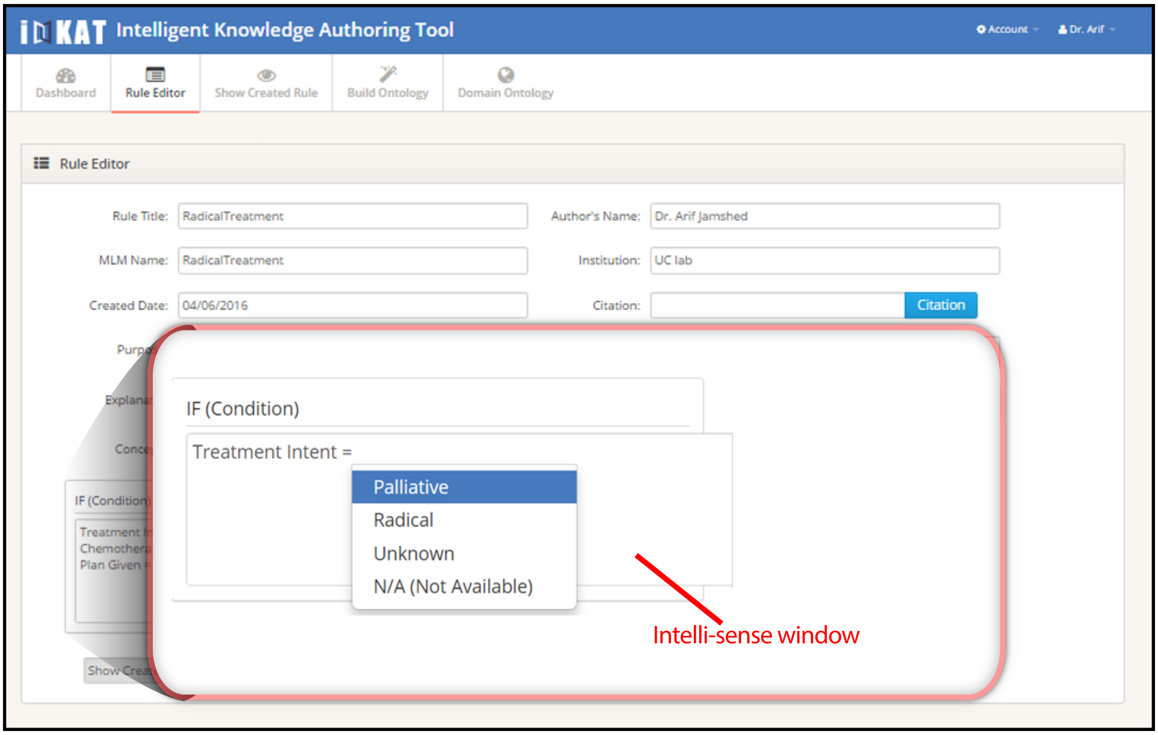

Figure A.3: Detailed view of Rule 1.

When the physician wants to save a created rule by clicking the *Save Rule* button, the corresponding MLM is generated in the back-end process. The generated MLM is represented in standard data model vMR concepts and SNOMED CT codes, instead of concepts in the understandable rule format on the user interface. After successful generation of the MLM, it is stored in the MLM knowledge base as text files and in the database repository in structured format. Physicians can see the newly created MLM by clicking the *Show Created MLM* button, and the result is displayed on the *Rule Viewer* page, as shown in Figure A.4.

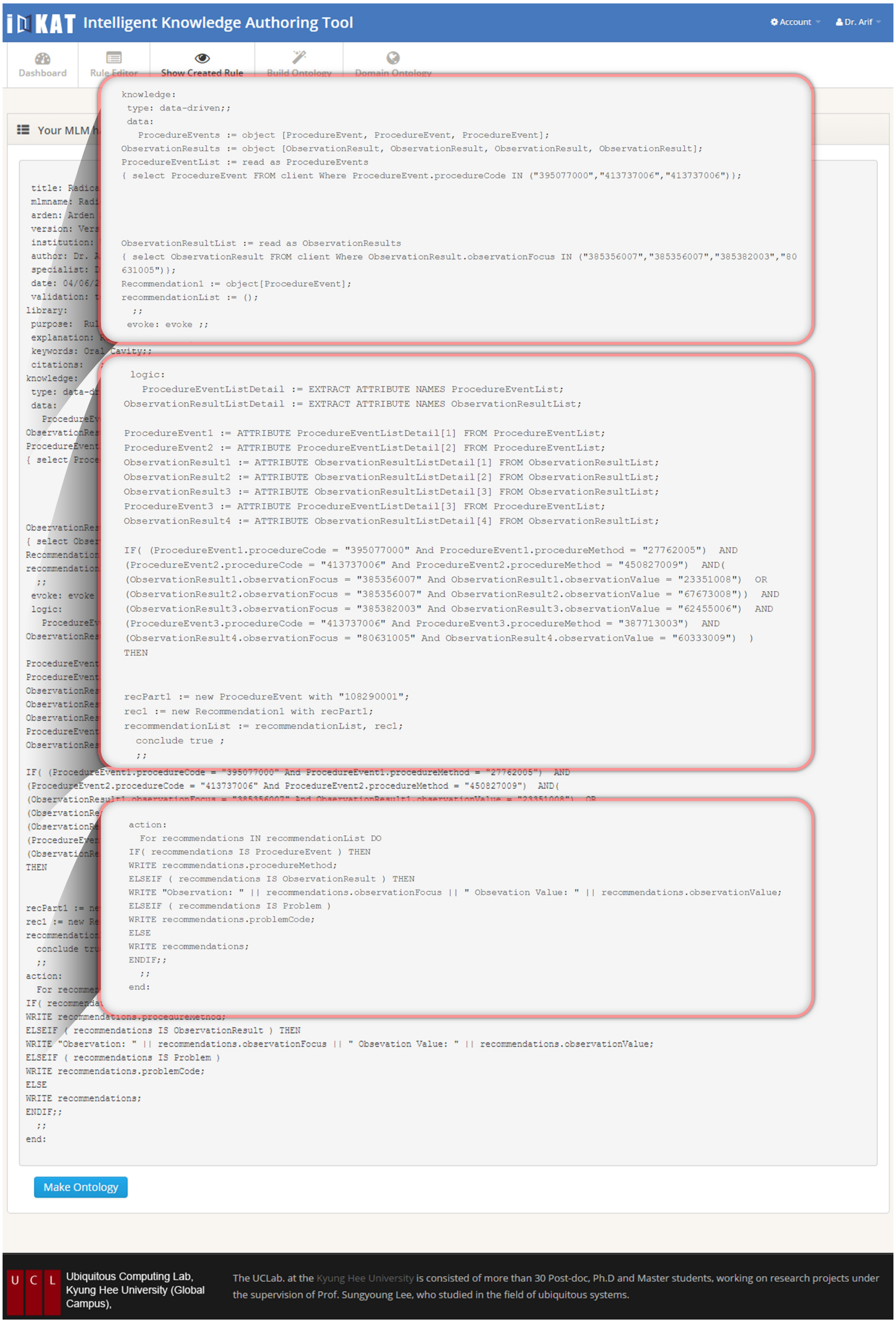

Figure A.4: MLM view for Rule 1.

A.2 Detailed explanation of the generated MLM

```
maintenance:
 title: RadicalTreatment;;
 mlmname: RadicalTreatment;;
 arden: Arden Syntax V2.7;;
 version: Version 2.7;;
 institution: UC lab;;
 author: Dr. Arif Jamshed;;
 specialist: Dr. Arif Jamshed;;
 date: 04/06/2015;;
 validation: testing;;
library:
 purpose:  Rule for radical patients treatment;;
 explanation: Rule for radical patients treatment;;
 keywords: Oral Cavity;;
 citations:   ;;
knowledge:
 type: data-driven;;
 data:
   ProcedureEvents := object [ProcedureEvent, ProcedureEvent, ProcedureEvent];
ObservationResults := object [ObservationResult, ObservationResult, ObservationResult,
    ObservationResult];
ProcedureEventList := read as ProcedureEvents
{ select ProcedureEvent FROM client Where ProcedureEvent.procedureCode IN ("395077000","
    413737006","413737006")};

ObservationResultList := read as ObservationResults
{ select ObservationResult FROM client Where ObservationResult.observationFocus IN ("
    385356007","385356007","385382003","80631005")};
Recommendation1 := object[ProcedureEvent];
recommendationList := ();
  ;;
 evoke: evoke ;;
 logic:
   ProcedureEventListDetail := EXTRACT ATTRIBUTE NAMES ProcedureEventList;
ObservationResultListDetail := EXTRACT ATTRIBUTE NAMES ObservationResultList;

ProcedureEvent1 := ATTRIBUTE ProcedureEventListDetail[1] FROM ProcedureEventList;
ProcedureEvent2 := ATTRIBUTE ProcedureEventListDetail[2] FROM ProcedureEventList;
ObservationResult1 := ATTRIBUTE ObservationResultListDetail[1] FROM
    ObservationResultList;
```

```
ObservationResult2 := ATTRIBUTE ObservationResultListDetail[2] FROM ObservationResultList;
ObservationResult3 := ATTRIBUTE ObservationResultListDetail[3] FROM ObservationResultList;
ProcedureEvent3 := ATTRIBUTE ProcedureEventListDetail[3] FROM ProcedureEventList;
ObservationResult4 := ATTRIBUTE ObservationResultListDetail[4] FROM ObservationResultList;

IF( (ProcedureEvent1.procedureCode = "395077000" And ProcedureEvent1.procedureMethod = "27762005")  AND
(ProcedureEvent2.procedureCode = "413737006" And ProcedureEvent2.procedureMethod = "450827009")  AND(
(ObservationResult1.observationFocus = "385356007" And ObservationResult1.observationValue = "23351008")  OR
(ObservationResult2.observationFocus = "385356007" And ObservationResult2.observationValue = "67673008"))  AND
(ObservationResult3.observationFocus = "385382003" And ObservationResult3.observationValue = "62455006")  AND
(ProcedureEvent3.procedureCode = "413737006" And ProcedureEvent3.procedureMethod = "387713003")  AND
(ObservationResult4.observationFocus = "80631005" And ObservationResult4.observationValue = "60333009")  )
THEN

recPart1 := new ProcedureEvent with "108290001";
rec1 := new Recommendation1 with recPart1;
recommendationList := recommendationList, rec1;
  conclude true ;
  ;;
action:
  For recommendations IN recommendationList DO
IF( recommendations IS ProcedureEvent ) THEN
WRITE recommendations.procedureMethod;
ELSEIF ( recommendations IS ObservationResult ) THEN
WRITE "Observation: " || recommendations.observationFocus || " Obsevation Value: " || recommendations.observationValue;
ELSEIF ( recommendations IS Problem )
WRITE recommendations.problemCode;
ELSE
WRITE recommendations;
ENDIF;;
  ;;
```

```
68 end:
```

Listing A.1: Created MLM for oral cavity cancer treatment plan of Rule 5

In the newly created MLM, all information is mapped and shown in the correct slots of the MLM. This information consists of *title*, *mlmname*, *institution*, *purpose*, and *explanation*. The *data* and *logic* slots are saved with integration of vMR concepts and SNOMED CT codes. In *data*, the system creates input objects of the vMR classes. These objects include three for *ProcedureEvent* for the DCM concept *Treatment Intent*, two for *Treatment Plan Given (Chemoinduction and Surgery)*, and four for *ObservationResult* for DCM concepts of *T (T1 and T2)*, one concept for *N*, and one concept for *Clinical Stage (S)*. The system requests input values for objects from the client as shown in lines 21-25 in MLM Listing A.1. Similarly, the system inputs the SNOMED CT codes (e.g., *395077000* for *Treatment Intent*, *413737006* for concept *Treatment Plan Given*, *413737006* for concept *T*, *385382003* for concept *N*, and *80631005* for concept *S*) in lines 22 and 25. Lines 31-32 show the declaration of two output recommendations; both of these recommendations belong to the vMR class *ProcedureEvent* for the *Treatment Plan*.

In *logic*, the system extracts values of *Treatment Intent, Treatment Plan Given, T, N,* and *S* from the lists of *ProcedureEvent* and *ObservationResult* shown in lines 34-40. Lines 42-54 show the *IF, THEN* part of the MLM. In *IF*, the *ProcedureEvent1.procedureCode = "395077000"* shows the *Treatment Intent* and *ProcedureEvent1. ProcedureMethod = "27762005"* shows the *Radical*. Likewise, *ObservationResult1.observationFocus = "385356007"* is used for clinical stage *T*, while *ObservationResult1.observationValue = "23351008"* shows the value *T1*. In the same manner, the key and values of facts *"T = T2," "N = N1," "Treatment Plan Given = Chemoinduction,"* and *"Treatment Plan Given = Surgery"* are generated.

In the *Then* part of *logic*, the system creates output object *"ProcedureEvent"* with SNOMED CT codes *"108290001"* for *Radiotherapy* in line 51. In the *action* slot of MLM, the system writes all generated output objects in lines 57-66 that were created for recommendation. All DCM, vMR, and SNOMED CT mappings are listed in Table 4.1.

A.3 Interpreted Requirements for the Authoring Tool

The following table shows the final list of selected and interpreted requirements according to the knowledge acquisition tool. The *Priority* column shows the requirement category (i.e., E for *Essential*, O for *Optional*, and R for *Recommended*). *Req. Number* column shows the requirement number. *Description* shows the requirement statement for CIMT. The *Interpretation for Knowledge Authoring Tool* column describes the requirement statement with respect to knowledge acquisition tools. The column *Implementation Methodology* shows the corresponding methodology to implement the requirement in our system.

Table A.1: **Classification and interpretation of requirements for CIMT to the Intelligent-Knowledge Authoring Tool.**

Priority	Req. Number	Description	Interpretation for Knowledge Authoring Tool	Implementation Methodology
E	R1	Be able to define clinical information models according to a defined technical specification for structuring clinical information in EHR systems.	**IR1**: Knowledge Authoring Tool able to create knowledge that is aligned with technical specifications for structuring the clinical information in EHR.	Our proposed system generates the MLMs using HL7 standard data model vMR for structuring and easy integrating with EHR systems.
E	R2	Support the semantic interoperability of EHR systems (Data Model, Std terminology).	**IR2**: Create knowledge support for easy integration with EHR workflows.	We are using standard the data model vMR and standard terminologies of SNOMED CT to enhance interoperability.
E	R3	Ensure consistency of information collected by enabling the definition of clinical information models generic enough to be compatible in multiple scenarios through specialization mechanisms for the additional constraints of each local scenario.	**IR3**: KAT should rely on and bind to local and standard clinical models and vocabulary for easy rule construction for localized recommendation interventions.	For MLM generation, the SRM maps DCM concepts with SNOMED CT and vMR. When new local concepts are added to the model, it maps them with corresponding vMR class and then with corresponding top-level concepts of SNOMED CT, which allows localized rule creation under the standard constraints.
E	R4	Definition and validation of the clinical information models according to a formal syntax.	**IR4**: KAT should support the validation process to ensure the validity of clinical rules.	We validate the created MLM with the structure and syntax of standard HL7 MLM.
E	R5	Import and export clinical information models according to the following formal syntaxes: XML and ADL.	**IR5**: KAT allows transformation of rules into multiple formats specified by the knowledge representation scheme.	Currently, our system presents knowledge in MLM format; in compilation module, we will represent it in ArdenML.

Continued on next page

Table A.1 – *Continued from previous page*

Priority	Req. Number	Description	Interpretation for Knowledge Authoring Tool	Implementation Methodology
E	R6	Represent data types according an accepted data type standard (e.g. ISO 21090 standard or a subset of this).	**IR6**: KAT should support standard data types according to the used standard data model for rule creation.	We are using standard datatypes of MLM that are standard for HL7 community.
E	R8	Provide an automatic parser for the defined clinical information model.	**IR8**: KAT includes parsers for different supported knowledge representation schemes.	Our system has parsers to re-read the created rules from MLM text files as well as from the knowledge base.
E	R9	Tools will verify that clinical information model and their instances are semantically and syntactically consistent.	**IR9**: KAT should semantically and syntactically validate rules according to the representation scheme.	According to SRM, our system validates the used concepts with vMR and SNOMED CT hierarchy, while validating the structure according to the HL7 MLM standard.
E	R10	The tool allows the author to create term bindings by connecting with Terminology Servers using (e.g. using CTS2) or another suitable terminology server communication specification.	**IR10**: KAT needs to bind concepts to standard terminologies to enhance shareability and and simplify integration.	Our system binds the rule editor with standard terminologies of SNOMED CT for easy selection of the desired concepts; internally, the MLM is generated with SNOMED CT codes of the corresponding concepts as well.
E	R11	Should include an intuitive graphical user interface for navigating large taxonomies.	**IR11**: KAT should manage a large number of rules and their dependencies in easy and understandable GUIs.	Our system provides DCM concepts in tree form for easy navigation and selection of concepts while large number of rules can be navigated through the provided dashboard.
E	R13	Should include mechanisms that enable users and find a clinical information models in the repository by searching on any of its structured information properties.	**IR13**: KAT allows easy interface for searching the large number of rules within the knowledge base.	Our system provides facility to search the existing rules in the knowledge base using the dashboard, while searchin the desired concept in the DCM tree using the defined category panels.
E	R14	Should export its clinical information model in at least one format that conforms to a published international standard or specification.	**IR14**: KAT should support at least one standard knowledge representation format.	Our system generates HL7 standard Arden Syntax MLM to share with other organizations.

Continued on next page

Table A.1 – *Continued from previous page*

Priority	Req. Number	Description	Interpretation for Knowledge Authoring Tool	Implementation Methodology
E	R16	Should allow collaborative authoring of clinical information models according to the established roles. As well as recording experts and organization participating in this process.	**IR16**: KAT should support roles to identify and manage the ownership of the created knowledge rules.	Our system provides facility for role management, i.e., each user has access to the knowledge base according to their roles.
E	R17	Should provide mechanisms to support multiple language translations of a clinical information model.	**IR17**: KAT should provide multilingual support for knowledge creation to cover maximum regions.	Currently, our system has no functionality to create MLMs in different languages.
E	R18	Should enable the formal definition of clinical content by domain experts without the need for technical understanding.	**IR18**: KAT allows abstraction to use localized concepts and enables automatic transformation of the underlying knowledge representation scheme while hiding the underlying technical complexity of concepts and syntax.	Our system provides abstraction to users for writing complex structure and syntax of MLM. The experts do not deal with complex structure of MLM and data model vMR.
E	R19	Should ensure the definition of purpose, appropriate description of usage, and precise mention of clinical information model domain.	**IR19**: KAT should ensure the meta information of each rule from the expert in self-explanatory manner.	The users can enter information about purpose, functionality, and other rule details using an easy-to-use interface. The system saves information in the maintenance slot of MLM.
O	R21	Facilitate the implementation of EHR systems that meet clinical requirements.	**IR21**: KAT should have standard conceptual models that enable easy integration of knowledge base with EHR workflows.	Implementation methodology for R1.
O	R45	Import/select the Reference Model that will lead underpin the definition.	**IR45**: The conceptual model used in KAT should be validated using standard reference model.	We selected vMR as the reference model, and it leads underpinning of the definition.
O	R48	Tools should suggest clinical information modelers with candidate terminology/ontology terms based on their semantic underlying model.	**IR48**: KAT should suggest candidate standard terminologies when experts write knowledge rules.	We provide SNOMED CT terminologies as standard.
O	R51	Should integrate or link to educational material to teach clinicians how to participate either in core and validation domain expert group.	**IR51**: Should integrate or link to educational material to teach clinicians how to participate in core and validation domain expert group.	Our system facilitates experts to link some educational material to the rules as evidence.

Continued on next page

Table A.1 – *Continued from previous page*

Priority	Req. Number	Description	Interpretation for Knowledge Authoring Tool	Implementation Methodology
O	R52	Should allow to assign or edit the GUI presentation capabilities for local purposes, making possible that clinician/administrator edit the local presentation.	**IR52**: KAT should allow edits to the GUI presentation and domain model according to the interest of experts.	We only change the GUI regarding the DCM tree with category-based selection.
R	R23	Support the organizational needs relating to the definition process, with coordination capabilities among clinical information modelling experts and clinical teams to provide a common or consensus agreed definition of the clinical information model.	**IR23**: KAT should support the organizational needs to create domain knowledge with the help and consensus of domain experts.	Our system provides a DCM concept as local concepts mapping with vMR and SNOMED CT in SRM. This model needs consensus and collaboration of clinical experts and knowledge engineers.
R	R25	Promote the clinician adoption with a simplified and guided view well understood by them that guide their participation in the modelling process.	**IR25**: KAT should provide simplified and guided views to the experts and should hide all complexity when writing knowledge rules.	Our system provides an easy to understand and well-organized editor to create knowledge that hides the complex syntax and structure of the MLM.
R	R26	Define semantic and syntactic patterns in the form of constraints to on the selected Reference Model.	**IR26**: KAT should bind rule authoring to the standard data models and vocabulary to fulfill the constraints of the reference model.	We provide abstraction to MLM with the vMR data model; therefore, the expert is restricted with regard to wrong classes or attributes of vMR.
R	R27	Provide an automatic testing environment for systems using the defined clinical information model.	**IR27**: Provide a testing environment to test the behavior of newly created knowledge before production.	Our system validates the MLM at runtime, either during testing or production. In the future, a testing environment will be provided.
R	R29	Should include visualization components for viewing complex term relationships.	**IR29**: Should include understandable and manageable components and views for domain experts to create knowledge in an easy manner.	Implementation methodology for R25.
R	R30	Should facilitate the use of the clinical information model to transform/map from existing data.	**IR30**: KAT should create knowledge rules with standard models and vocabulary to support the existing data of organizations.	Implementation methodology for R2.
R	R31	Should allow to define transformations of the clinical information models to/from other specifications.	**IR31**: KAT should allow transformation of the knowledge rules into different formats of knowledge representation schemes.	Implementation methodology for R5.

Continued on next page

Table A.1 – *Continued from previous page*

Priority	Req. Number	Description	Interpretation for Knowledge Authoring Tool	Implementation Methodology
R	R32	A repository service should provide a notification service to experts and systems about clinical information model updates, additions and backwards compatibility.	**IR32**: The system should provide a notification service to experts and administrators about rules updates, additions, and backward compatibility.	According to role management, whenever model or knowledge rules are changed, it will notify the persons of concern.
R	R33	Where more than one format is supported, requester user or system will be able to nominate the preferred retrieval format.	**IR33**: The tool allows the transformation of knowledge rules into multiple representation formats for retrieval of knowledge according to expert interest.	Our system facilitates retrieval of the rules in the desired format.
R	R34	Requesters of obsolete versions of an clinical information model shall be provided with a notification that an update (or updates) exist and be able to nominate the version(s) to be returned.	**IR34**: According to role management, the experts should be notified about updates in the knowledge rules and be able to nominate the correct updated version of the knowledge rule.	Implementation methodology for R32.
R	R37	Should provide mechanisms to assign the following roles to experts participating in the clinical information modelling process and document this information in the final clinical information model produced: editor,author and reviewer.	**IR37**: In KAT, the three main roles of editor, author, and reviewer should exist, each of which should be able to process the knowledge rules.	Implementation Methodology for R32.
R	R39	Should provide the means to define the clinical and usage scope of the clinical information model in a structured and coded format, in order to be able to check for possible scope overlap with other clinical information model.	**IR39**: The system should provide a mechanism based on standard data model and vocabulary to resolve merging conflicts between two knowledge bases.	Our system gives an immediate prompt to the expert when the logic part of a rule overlaps with existing knowledge rules during creation of rules and merging with other knowledge bases.
R	R40	Should implement clinician understandable mechanisms for a guided process for local specialization and validation purposes.	**IR40**: KAT should implement understandable and guided mechanisms for the clinicians to adapt localized rules according to the standard data model.	Implementation methodology for R39.

Continued on next page

Table A.1 – *Continued from previous page*

Priority	Req. Number	Description	Interpretation for Knowledge Authoring Tool	Implementation Methodology
R	R41	Should be able to create prototype screens for domain expert validation of the defined clinical information model auto-generates example GUIs to test the creation of example instances.	**IR41**: KAT should provide GUI screens to test the rule validation with real data.	Will be implemented in the future.
R	R42	User friendly interface for clinicians including drag and drop capabilities to be able to manage multiple clinical information models easily.	**IR42**: User-friendly interface for clinicians including drag and drop/IntelliSense functionalities to manage knowledge rules in an easy way.	Our system provides a user-friendly interface for rule creation with IntelliSense functionality and drag and drop mechanism of concept selection from the DCM tree.
R	R43	Editorial role can examine changes, and accept or reject changes.	**IR43**: Editorial role should examine the created/ updated knowledge rules.	Implementation Methodology for R32.
R	ER57		**ER57**: Provide DCM in hierarchical form for easy selection of required concepts during knowledge creation.	Our system provides a DCM tree that contains all understandable domain concepts used in local HMIS systems.
R	ER58		**ER58**: The knowledge editor should provide contextual selection of a required value of a concept from the value set using the IntelliSense window.	Our system facilitates physician selection of the desired concepts from the IntelliSense window during rule creation, which populate from DCM concepts or SNOMED CT concepts that depends on the experts' choice.

Appendix B

List of Publications

International Journal Papers:

[1] **Taqdir Ali**, Maqbool Hussain, Wajahat Ali Khan, Muhammad Afzal, Jamil Hussain, Rahman Ali, Waseem Hassan, Arif Jamshed, Byeong Ho Kang and Sungyoung Lee, ”Multi-model-based interactive authoring environment for creating shareable medical knowledge”, Computer methods and programs in biomedicine, (SCI, IF:2.503), 150 (2017): 41-72.

[2] **Taqdir Ali**, Maqbool Hussain, Muhammad Afzal, Wajahat Ali Khan, Taeho Hur, Muhammad Bilal Amin, Dohyeong Kim, Byeong Ho Kang and Sungyoung Lee, ” Clinically Harmonized Wellness Concepts Model for Health and Wellness Services”, IEEE Access, (SCIE, IF: 3.244), Vol.6, pp.26660-26674, 2018.

[3] Muhammad Afzal, Syed Imran Ali, Rahman Ali, Maqbool Hussain, **Taqdir Ali**, Wajahat Ali Khan, Muhammad Bilal Amin, Byeong Ho Kang, and Sungyoung Lee, ”Personalization of Wellness Recommendations Using Contextual Interpretation”, Expert Systems with Applications (IF: 3.928), Vol.96, pp.506-521, 2018.

[4] Rahman Ali, Muhammad Afzal, Muhammad Sadiq, Maqbool Hussain, **Taqdir Ali**, Asad Masood Khatak, Do Hyung Kim and Sungyoung Lee , ”Knowledge-Based Reasoning and Recommendation Framework for Intelligent Decision Making”, Expert Systems (SCIE, IF: 1.18), doi:10.1111/exsy.12242, 2018

[5] Muhammad Afzal, Maqbool Hussain, Wajahat Ali Khan, **Taqdir Ali**, Sungyoung Lee, Eui-Nam Huha, Hafiz Farooq Ahmad, Arif Jamshed, Hassan Iqbal, Muhammad Irfan and Manzar

Abbas Hydari, "Comprehensible knowledge model creation for cancer treatment decision making", Computers in Biology and Medicine, (SCI, IF:1.521), Vol.82, pp.119-129, 2017.

[6] Muhammad Afzal, Maqbool Hussain, Wajahat Ali Khan, **Taqdir Ali**, Arif Jamshed and Sungyoung Lee, "SEAS: Smart Extraction and Analysis System for Clinical Research", Telemedicine and e-Health (SCI, IF: 1.79), doi:10.1089/tmj.2016.0157, 2017

[7] Muhammad Bilal Amin, Oresti Banos, Wajahat Ali Khan, Hafiz Syed Muhammad Bilal, Jinhyuk Gong, Dinh-Mao Bui, Soung Ho Cho, Shujaat Hussain, **Taqdir Ali**, Usman Akhtar, Tae Choong Chung and Sungyoung Lee, "On Curating Multimodal Sensory Data for Health and Wellness Platforms", Sensors (SCIE, IF: 2.033), vol. 16,no. 7, doi:10.3390/s16070980 , 2016

[8] Maqbool Hussain, Muhammad Afzal, **Taqdir Ali**, Rahman Ali, Wajahat Ali Khana, Arif Jamshed, Sungyoung Lee, , Byeong Ho Kang and Khalid Latif, "Data-driven knowledge acquisition, validation, and transformation into HL7 Arden Syntax", Artificial Intelligence in Medicine (SCI, IF:2.109), DOI:10.1016/j.artmed.2015.09.008, 2015.

[9] Muhammad Afzal, Maqbool Hussain, **Taqdir Ali**, Jamil Hussain,Wajahat Ali Khan, Sungyoung Lee and Byeong Ho Kang, "Knowledge-Based Query Construction Using the CDSS Knowledge Base for Efficient Evidence Retrieval", Sensors (SCIE, IF:2.245), Vol.15, Issue 9, pp.21294-21314, 2015.

[10] Rahman Ali, Muhammad Hameed Siddiqi, Muhammad Idris Ahmed, **Taqdir Ali**, Shujaat Hussain, Eui-Nam Huh, Byeong Ho Kang and Sungyoung Lee, "GUDM: Automatic Generation of Unified Datasets for Learning and Reasoning in Healthcare", Sensors (SCIE, IF: 2.245), Vol.15, No.7, pp.15772-15798, 2015.

[11] Maqbool Hussain, **Taqdir Ali**, Wajahat Ali Khan, Muhammad Afzal, Sungyoung Lee and Khalid Latif, "Recommendations Service for Chronic Disease Patient in Multimodel Sensors Home Environment", Telemedicine and e-Health (SCI, IF:1.668), Vol.21, No.3, pp.185-199, 2015.

[12] Asad Masood Khattak, Noman Akbar, Mohammad Aazam, **Taqdir Ali**, Adil Mehmood Khan, Seokhee Jeon, Myunggwon Hwang and Sungyoung Lee, ”Context Representation and Fusion: Advancements and Opportunities”, Sensors (SCIE, IF:2.048), 14(6), 9628-9668, DOI:10.3390/s140609628, 2014.

International Conference Papers:

[13] **Taqdir Ali**, and Sungyoung Lee, ”Reconciliation of SNOMED CT and domain clinical model for interoperable medical knowledge creation”, 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2017), Jeju island, Korea, July 11-15, 2017.

[14] **Taqdir Ali**, and Sungyoung Lee, ”Wellness Concepts Model use and effectiveness in intelligent knowledge authoring environment”, 14th International Conference on Smart Homes and Health Telematics (ICOST). Inclusive Smart Cities and Digital Health, pp. 271-282, Wuhan, China, May 25-27, 2016.

[15] **Taqdir Ali**, and Sungyoung Lee, ”Semantic Reconciliation Model for interoperable and shareable knowledge authoring environment”, 2016 International Symposium on Perception, Action and Cognitive Systems (PACS), Seoul, Korea, Oct 27-28, 2016.

[16] **Taqdir Ali**, Maqbool Hussain, Wajahat Ali Khan, Muhammad Afzal and Sungyoung Lee , ”Customized clinical domain ontology extraction for Knowledge Authoring Tool”, IMCOM (CUIMC) 2014, Siem Reap, Cambodia, Jan 9-11, 2014.

[17] **Taqdir Ali**, Maqbool Hussain, Wajahat Ali Khan, Muhammad Afzal, Byeong Ho Kang and Sungyoung Lee, ”Arden Syntax Studio: Creating Medical Logic Module as Shareable knowledge”, INISTA 2014, Alberobello, Italy, Jun 23-25, 2014.

[18] **Taqdir Ali**, Maqbool Hussain, Wajahat Ali Khan, Muhammad Afzal and Sungyoung Lee, ”Authoring Tool: Acquiring sharable knowledge for Smart CDSS”, 35th IEEE Annual International Conference of the Engineering in Medicine and Biology (EMBC 2013), Osaka, Japan, July 3-7, 2013.

[19] Muhammad Afzal, Maqbool Hussain, Wajahat Ali Khan, **Taqdir Ali** and Sungyoung Lee, ”Towards Evidence Adaptive Clinical Decision Support System”, 12th International Conference on Ubiquitous Healthcare (u-Healthcare 2015), Osaka, Japan, Nov 30- Dec 2, 2015.

[20] Wajahat Ali Khan, Muhammad Bilal Amin, Oresti Banos, **Taqdir Ali**, Maqbool Hussain, Muhammad Afzal, Shujaat Hussain, Jamil Hussain, Rahman Ali, Maqbool Ali, Dongwook Kang, Jaehun Bang, Tae Ho Hur, Bilal Ali, Muhammad Idris, Asif Razzaq, Sungyoung Lee and Byeong Ho Kang, ”Mining Minds: Journey of Evolutionary Platform for Ubiquitous Wellness”, 12th International Conference on Ubiquitous Healthcare (u-Healthcare 2015), Osaka, Japan, Nov 30- Dec 2, 2015.

[21] Wajahat Ali Khan, Muhammad Idris, **Taqdir Ali**, Rahman Ali, Shujaat Hussain, Maqbool Hussain, Muhammad Bilal Amin, Asad Masood Khattak, Yuan Weiwei, Muhammad Afzal, Sungyoung Lee and Byeong Ho Kang, ”Correlating Health and Wellness Analytics for Personalized Decision Making”, 12th International Conference on Ubiquitous Healthcare (u-Healthcare 2015), Osaka, Japan, Nov 30- Dec 02, 2015

[22] Claudia Villalonga, Oresti Banos, Wajahat Ali Khan, **Taqdir Ali**, Muhammad Asif Razzaq, Sungyoung Lee, Hector Pomares and Ignacio Rojas, ”High-Level Context Inference for Human Behavior Identication”, 7th International Work-conference on Ambient Assisted Living (IWAAL 2015), Puerto Varas, Patagonia, Chile, Dec 2-3, 2015.

[23] Oresti Banos, Muhammad Bilal Amin,Wajahat Ali Khan, **Taqdir Ali**, Muhammad Afzal and Byeong Ho Kang, ”Mining Minds: an innovative framework for personalized health and wellness support”, 9th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth 2015), Trento, Italy, May 20-23, 2015.

[24] Oresti Banos, Jaehun Bang, Taeho Hur, Muhammad Hameed Siddiqi, Huynh-The Thien, Le-Ba Vui, Wajahat Ali Khan, **Taqdir Ali**, Claudia Villalonga and Sungyoung Lee, ”Mining Human Behavior for Health Promotion”, 37th Annual International Conference of The IEEE Engineering in Medicine and Biology Society, Aug 25-29, 2015.

[25] Oresti Banos, Muhammad Bilal Amin, Wajahat Ali Khan, Muhammad Afzel, Mahmood Ahmad, Maqbool Ali, **Taqdir Ali**, Rahman Ali, Muhammad Bilal, Manhyung Han, Jamil Hussain, Maqbool Hussain, Shujaat Hussain, Tae Ho Hur, Jae Hun Bang, Thien Huynh-The, Muhammad Idris, Dong Wook Kang, Sang Beom Park, Hameed Siddiqui, Le-Ba Vui, Muhammad Fahim, Asad Masood Khattak, Byeong Ho Kang and Sungyoung Lee, ”An Innovative Platform for Person-Centric Health and Wellness Support”, 3rd International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO 2015), Granada, Spain, April 15-17, 2015

[26] Muhammad Afzal, Maqbool Hussain, **Taqdir Ali**, Wajahat Ali Khan, Sungyoung Lee and Byeong Ho Kang, ”MLM-Based Automated Query Generation for CDSS Evidence Support”, UCAmI 2014, Belfast, Ireland, Dec 2-5, 2014.

[27] Muhammad Afzal, Maqbool Hussain, Wajahat Ali Khan, **Taqdir Ali**, Sungyoung Lee and Byeong Ho Kang, ”Knowledge Button: An Evidence Adaptive Tool for CDSS and Clinical Research”, INISTA 2014, Alberobello, Italy, Jun 23-25, 2014.

[28] Muhammad Afzal, Maqbool Hussain, Wajahat Ali Khan, **Taqdir Ali**, Sungyoung Lee and Hafiz Farooq Ahmad, ”Meaningful Integration of Online Knowledge Resources with Clinical Decision Support System”, 11th International Conference On Smart homes and Health Telematics(ICOST 2013), Singapore, June 19-21, 2013.

[29] Maqbool Hussain, **Taqdir Ali**, Muhammad Afzal, Wajahat Ali Khan, Sungyoung Lee, “Let’s Fire the CDSS integration issue with FHIR”, 14th International HL7 Interoperability Conference (IHIC 2013) and FHIR Connectathon, Sydney, Australia, 27-29 October, 2013.