Computerizing Clinical Pathways: Ontology-Based Modeling and Execution

Ali DANIYAL, Samina Raza ABIDI and Syed Sibte Raza ABIDI[1]
NICHE Research Group, Faculty of Computer Science, Dalhousie University, Canada

Abstract. Clinical Pathways (CP) stipulate an evidence-based patient care workflow for a specific disease within a localized setting. We present an ontology-based approach for computerizing CP so that they can be executed at the point-of-care. We present our CP modeling approach that features the integration of multiple localized CP to realize a unified disease-specific CP. The execution of the ontological CP model is achieved by our property abstraction method that assigns functional behaviors to existing semantic properties to facilitate their execution. Using our methods we have developed a prostate cancer management system.

Keywords. Clinical Pathway, Decision Support, Ontology, Prostate Cancer

1. Introduction

In Canada, the management of prostate cancer follows an integrated care approach involving multiple medical disciplines, care settings and health professionals. This integrated approach demands an effective coordination between family physicians, urologists, radiation oncologists and nurses to manage the patient's care trajectory. One possible solution to coordinate the care activities is to subscribe to Clinical Pathways (CP) as they are evidence-based patient care algorithms/charts that describe the care process for specific medical conditions within a localized setting [1]. However, this raises two practical issues: (a) the *development* of 'pragmatic' prostate cancer CP that determines the sequence of care activities in keeping with the resource realities; and (b) the *execution* of the CP at the point of care as part of the clinical workflow [2].

In this paper we present a prostate cancer care planning and management system based on Semantic Web technologies. We describe the three phases of the project: (i) The knowledge engineering phase involved the development of three location-specific prostate cancer CP for three different Canadian cancer care institutions in Halifax, Winnipeg and Calgary; (ii) The knowledge modeling phase involved the semantic modeling of the CP knowledge, in terms of an OWL ontology, leading to the computerization of the CP. The feature of our modeling approach is that it not only models the three different location-specific CP within a unified structure, but it also allows the 'merging' of these location-specific CP along common processes, actions and recommendations; and (iii) the execution phase involved the development of a logic-based execution engine that uses an innovative *property abstraction* technique that allows the ontologically-modeled CP to be executed with patient data. The

[1] Corresponding Author: Syed Sibte Raza Abidi, Faculty of Computer Science, Dalhousie University, Halifax, NS, B3H 1W5, Canada. Email: sraza@cs.dal.ca

execution of the CP guides both the care-providers and the patient through the care trajectory.

2. Knowledge Engineering Phase: Developing a Prostate Cancer CP

This phase involved working with oncologists, urologists and nursing experts to elicit the underlying clinical processes for prostate cancer management in their respective institutions. A systematic analysis yielded three location-specific prostate cancer CP in terms of a workflow comprising four components—actions, decisions, branching and merging nodes and recommendations/plans. Each CP was divided into four consultations—namely (1) visit to family physician, (2) visit to primary urologist, (3) visit to secondary urologist and (4) treatment option. In each consultation a set of tasks are performed by a health professional to achieve a defined outcome. It was interesting to note that the three CP exhibited a significant deal of overlap at the task-level.

3. Knowledge Modeling Phase: Developing a Prostate Cancer CP Ontology

Knowledge modeling was done using semantic web techniques, in particular ontologies. Using the three location-specific CP, we developed an OWL based *prostate cancer CP ontology* (PC Ontology) that represents the diagnostic, treatment and operational concepts within the CP, and relates these concepts using semantic and clinical pragmatic relationships. We introduced unique knowledge constructs that allow the 'merging' of the three CP, based on commonalities of tasks across multiple institutions, to realize a unified ontological model for prostate cancer CP (see Figure 1a). The CP model aggregates the common steps across the multiple CP and represents them as common path. However, when a location-specific CP performs a unique task then we use a *branching node* to allow the CP to branch off the main CP path. To allow the branched CP to integrate with the main CP path, if it shares a common task with the other CP, we use a *merging node* that allows multiple branches to merge to realize a common path—in figure 1b Consult_4 is a merging node.

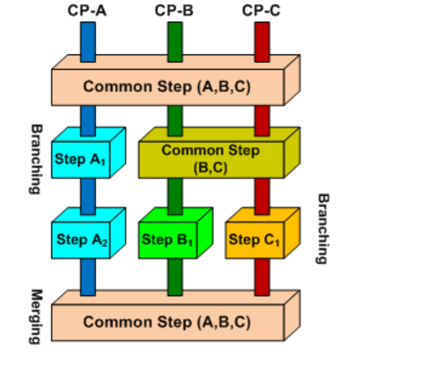


Figure 1.: A schematic of a unified CP for three different sites (A, B, C), highlighting both branching and merging nodes

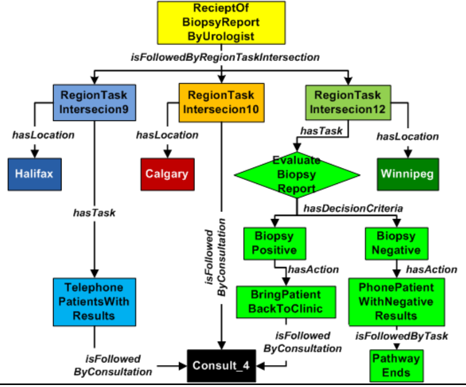


Figure 1b. The branching node 'ReceiptOfBiopsyReport' spawns three location-specific branches that merge at the merging node 'Consult 4'

The complete details of the PC Ontology is provided in [3]. Here we highlight the modeling of key concepts as CLASSES and *relationships*. Our ontology begins with class PLAN which corresponds to all four consultations. DECISION-CRITERIA models the choices, for instance the results of an INVESTIGATION, that determines the next step which is represented as a TASK that is further classified as CONSULTATION-TASK, NON-CONSULATION-TASK, REFERRAL-TASK and FOLLOW-UP-TASK. We have modeled a large number of relationships between classes; here we present some salient relationships. PLAN, TEST-RESULT and PATIENT-CONDITION-SEVERITY have relationship *isFollowedByTask* with TASK to model the situation whereby when the PATIENT-CONDITION-SEVERITY is `NonUrgent' then the next task modeled by the relationship *isFollowedByTask* will be `BiopsyIsNotBooked'. Intervals between TASK, TREATMENT and FOLLOW-UP are modeled by the relationship *hasInterval* with INTERVAL-EVENT.

The modeling of branching and merging nodes is explained below. Branching involves a location-specific CP diverging from the unified path because the next tasks at a specific location differ with those at other locations. Branching nodes are modeled as an intersection between two classes to represent a unique individual that is the function of two intersected classes, such as REGION-TASK-INTERSECTION, represents an intersection between REGION and TASK to signify a unique task performed at a particular region. The merging node allows unique CP branches, corresponding to different locations, to merge at a common task (see figure 1b).

4. Execution Phase: Developing a CP Execution Engine

The execution phase involved converting the PC Ontology to an executable format and then developing an execution engine to execute the CP with patient data. The execution of a CP can be understood as the traversal of a workflow or a state graph where each state contains two elements: (a) actions to be performed whilst satisfying any local constraints, and (b) the potential next state(s). We explain below the methods used to develop the CP execution engine.

To execute the CP ontology we need to view it as a *state-graph* comprising *states* (similar to nodes) that refer to 'instances' of a CLASS and edges that depict the *properties* (or *relationships)* between the states. The ordering of the states is determined by the CP's workflow and a traversal of the state-graph, based on satisfying a state's constraints, results in the execution of the CP. The state graph is represented as a set of triples such as s1 p s2 (where s1and s2 are states and p is a property). For example s1 is an instance of the class PLAN with value 'Consult1_ConsultationWithFamilyPhysician'; s2 is an instance of the class TASK and has the value 'PerformPSA_and_DRE_Test'; s1 is related to s2 by the property *hasTask* which results in a simple graph in which PLAN is followed by a TASK. To execute the CP we determined a state graph from the CP ontology as follows:

4.1. Property Abstraction

Ontology-Based knowledge modeling allows the classification of domain concepts into classes, whereas the properties present a conceptualization of relationship between classes. We explained earlier that the in a state graph the workflow is captured through a variety of domain-specific properties (or relationships). To design a CP execution engine, leveraging pre-defined properties within an ontology, we developed a new approach—termed as *Property Abstraction*—to streamline the use of the different properties for CP execution. Based on their behavior, the properties are distinguished as (a) *executional properties* that are used in the state graph to execute the CP; and (b) *informational properties* that provide information to the user for a particular state.

Our property abstraction approach involves: (a) *identifying* the similarities between the functional behavior of different properties to define high-level property classes. For example some properties have the behavior of relating two states such that one state follows the other, some properties represent decisions made at a state; (b) *classifying* the properties into property classes; and (c) *assigning* a specific workflow behavior to a property class that will be used by the execution engine to determine its actions, for example the STATE_CHANGE class of properties will manifest the behavior of moving from one state of the graph to another state. The property abstraction approach divided the 25 properties into 4 classes for executional properties—namely STATE_CHANGE_PROPERTY (6), STATE_INFORMATION _PROPERTY (2), DECISION_PROPERTY (2) & CONSTRAINT_PROPERTY (3)—and 3 classes for information properties—namely INFORMATION_PROPERTY (4) TIME_PROPERTY (5), LOCATION_SPECIFIC_INFORMATION_PROPERTY (3).

4.2. Extracting State Graph using Property Abstraction

Next we use the abstracted property classes to extract a state graph from the CP ontology. A state graph is a simplified model inspired by many existing models in literature that capture states in a system along with relationships among them, e.g. Finite State Machines [4] and UML State diagrams. A state graph is defined as a 5-tuple (C,P,S,E,D,L) where; C is a set of institution labels, P is a set of decision property labels, P is set of states, $E \subseteq S \times S$ is the set of unlabeled edges, $D \subseteq S \times S \times P$ is the set of decision edges and $L \subseteq S \times S \times C$ is the set of institution labeled edges. From the RDF graph of the CP ontology we extract the state graph by classifying the properties of the ontology in terms of E, D and L, where E = _CHANGE_PROPERTY, STATE_INFORMATION; D = DECISION_PROPERTY and L = CONSTRAINT_PROPERTY.

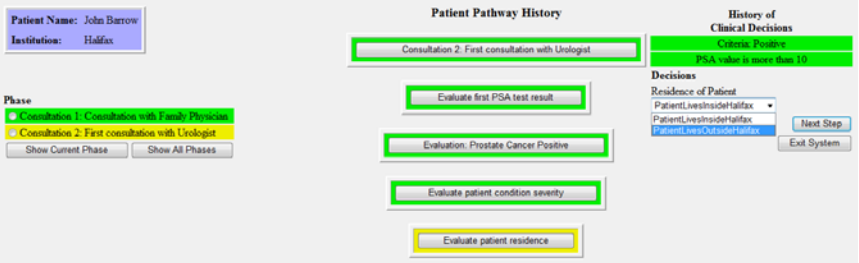
4.3. Parameterized Execution of Prostate Cancer CP

After extracting the CP's state graph from the PC ontology we use it to execute the CP. The parameter for the execution of the CP is an instance of the class PATHWAY_REGION that determines the location of the patient, and helps to select the corresponding CP which in essence is a parameterized sub-graph of the state graph. For a parameter value $v \in \text{PathwayRegion}$, the parameterized sub-graph of a state graph (C,P,S,E,D,L) is defined as (C,P,S_V,E,D,L_V) where $L_V=\{(x,y,v)\in L\}$ and $S_V=\{s\in S|\exists p\in \text{Plan}$ such that there is a path from p to x in $(C,P,S,E,D,L_V)\}$. The algorithm generates a set of decisions corresponding to each input current state, and based on the outcome of the decision it determines the next states. The entire CP is executed by moving from one state to another.

5. Prostate Cancer Care Management System

We developed a Prostate Cancer Care Management System using the PC Ontology and the execution engine described earlier. The system comprises four layers namely;

Interface, Execution, Property Abstraction and Ontology. Functionally, the system coordinates the care activities as per the location specific CP and stores the current treatment record for each patient. The system shows the location-specific care pathway for each patient as a sequence of states that are highlighted as completed states, active state and the next state. For each state, the execution engine performs the following: gathers and records patient data from the user/medical record, makes a decision, satisfies local constraints, provides related information/notification to the user. The CP execution is divided as per the four consultations and the user can either view just any specific consultation or the entire CP. The system has a web-based interface that is dynamically generated using the *hasLabel* property in the CP ontology and it shows (a) completed states in green, active states in yellow and by clicking on the active state the next states are displayed; (b) history of clinical decision made; and (c) values for the various decision points (as shown in figure 2).



Patient Name: John Barrow
Institution: Halifax

Phase

○ Consultation 1: Consultation with Family Physician
○ Consultation 2: First consultation with Urologist

Patient Pathway History

Consultation 2: First consultation with Urologist
Evaluate first PSA test result
Evaluation: Prostate Cancer Positive
Evaluate patient condition severity
Evaluate patient residence

History of Clinical Decisions

Criteria: Positive
PSA value is more than 10

Decisions

Residence of Patient: PatientLivesInsideHalifax

Figure 2. Screenshot of the prostate cancer management system. The left section shows consultations, the middle shows the patient's pathway and the right shows the decision made and decision options.

6. Concluding Remarks

We presented a CP modeling approach that allows the merging of specialized CP into a common model, whilst ensuring their location/task specific uniqueness. The concept of branching and merging nodes has been around but we were able to demonstrate it both at the modeling and execution level with a real health problem. We demonstrated the execution of clinical pathways modeled in terms of ontologies. Our execution approach is both generic and scalable such that it transforms an existing ontologically-modeled CP into an executable CP by providing a workflow-specific interpretation to the properties viz. our property abstraction approach. The execution of the prostate cancer CP was tested on a number of clinical scenarios of different complexity levels, and in each case the outcome was as per the original CP. The web-based system is available to health practitioners at their workplace—they can (i) access the patient's care record showing the completed care activities; and then (ii) administer the care process as per the CP. We are now planning system deployment at the three health institutions.

References

[1] L. de Bleser, R. Depreitere, K. de Waele, K. Vanhaecht, J. Vlayen, and W Sermeus, Defining pathways, *Journal of Nursing Management* **14** (2006), 553–563.

[2] S Chu, Reconceptualising CP system design, *Collegian* **8** (2001), 33–36.

[3] S. Abidi, S.S.R Abidi, S Hussain and L Butler, Ontology-Based modeling and merging of institution-specific prostate cancer clinical pathways. *Knowledge Management for Healthcare Processes Workshop at 18th European Conference on Artificial Intelligence*, Patras (Greece), July 21-25 2008.

[4] C. Cassandras, S. Lafortune, *Introduction to Discrete Event Systems*, Kluwer, 1999.