Translation Protégé
Knowledge for Executing
Clinical Guidelines

Jeong Ah Kim,
BinGu Shim, SunTae Kim, JaeHoon Lee, InSook Cho, Yoon Kim
Agenda

1. Motivation
2. How to translate
3. Implementation and Case study
4. Conclusion
Motivation

Definition of CDSS

- any piece of software that takes as input information about a clinical situation and that produces as output inferences that can assist practitioners in their decision making and that would be judged.

CDSS can

- give specific reminders at particular clinical situations
- give exact information to support drug choosing, dosing, preventing adverse drug effects
- support the health care management at the hospital level
- be used as educational systems for medical students or young doctors
In CDSS, core component is guidelines.
- Computer-interpretable guidelines (CIG) have been developed for decision support during clinical process
- Evidence based guideline practice promises to improve health care quality.

Several approaches for modeling the clinical guideline
- Arden syntax, EON, PRODIGY, GUIDE, GLIF,
- SAGE (Standard-based Sharable Active Guideline Environment)
SAGE

- uses standardized components that allow interoperability of guideline execution elements
- Integrate guideline-based decision support with the workflow of care process
- synthesizes prior guideline modeling work for encoding guideline knowledge
- *A Suite of Models and Services to Support Guideline Modeling and Execution*
- *Deployment-Driven Knowledge-Base Development Process*
- there is not publically available execution engine yet
Motivation

- **EHR Knowledge Engine**

CDSS Application

- Workflow Engine
- MQ Processor
- Rule Engine Adaptors

- Repository Manager
- Rule Executor
- Medical Function Lib

- Rule/Process Repository
- DBMS / FILE

- HTTP
- XML-RPC
- RMI

- Client API

- Java Virtual Machine

U-BRAIN
**Knowledge Model of u-BRAIN**

- Ontology-based
  - Domain Ontology defines the concepts and criterion value in each domain
  - Interface ontology define the required information from outside (ex: patient information stored in CIS)
  - Rule is defined to make the decisions with concepts in domain ontology and values in interface ontology
  - Each rule has identifier

- Structured workflow based

---

**Motivation**
How to translate

- **Our approach**
  - Analyze the SAGE representation formalism
  - Use protégé KnowledgeBase interface to get the SAGE object model
  - Apply “Export” plug-in development method to integrate SAGE model and u-BRAIN converter and u-BRAIN execution engine
    - SAGE object(Knowledge base) -> uEngine Object mapping -> serialize -> Plug-in Export -> XPD & XML for u-BRAIN representation
## How to Translate

- **Object model of SAGE and mapping to uBRAIN object**

<table>
<thead>
<tr>
<th>SAGE model</th>
<th>Meaning</th>
<th>features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guideline</td>
<td>Collection of associated Recommendation Set</td>
<td>Process</td>
</tr>
<tr>
<td>Recommendation Set</td>
<td>Decision Map or set of activity graph where the recommendations associated with the Context node is applicable.</td>
<td>Sub-Process</td>
</tr>
<tr>
<td>Context</td>
<td></td>
<td>Rule</td>
</tr>
<tr>
<td>Expression</td>
<td>expression language that can be used with any object-oriented model.</td>
<td>Rule</td>
</tr>
<tr>
<td>Concept</td>
<td>Constant atomic term</td>
<td>Rule</td>
</tr>
<tr>
<td>Variable</td>
<td>Meaningful result from executing the internal logic</td>
<td>Rule</td>
</tr>
<tr>
<td>Evidence Statement</td>
<td>represents a relationship between clinical conditions and interventions and additional contextual information and supporting references</td>
<td>Rule/Process</td>
</tr>
<tr>
<td>Activity graph</td>
<td>inter-related activities.</td>
<td>Process</td>
</tr>
<tr>
<td>Action</td>
<td>flow-of-control information</td>
<td>Process</td>
</tr>
<tr>
<td>Decision</td>
<td>representation of decision knowledge required to recommend a choice among alternatives</td>
<td>Rule/Process</td>
</tr>
</tbody>
</table>
How to Translate

- New Architecture of u-BRAIN
How to translate

Workflow at runtime

1. Get some initial basic data of specific patient and make initial interface XML
2. Execute knowledge
3. [if necessary) get mode data and add to interface XML
4. Execute VMR_query
5. Return queried data in interface XML
6. Return recommendation so on
7. Display the result
8. Store the result
How to translate

✓ **SAGE Workflow to u-BRAIN activity**
  - Each action node is mapped to one activity node
  - Decision node is mapped to also u-BRAIN activity to invoke rule engine to do decision-making using rule
  - Complex action node is mapped one decision making node and decision structure of activity

✓ **SAGE decision to u-BRAIN rule**
  - Each expression is mapped to rule expression (if then else)
  - Generate the interface model to access the EMR (external data resource)
How to Translate

2 Kinds Expression in translation perspectives

- EMR database access is not required during rule execution
  - N-ary criterion, variable_comparison_criterion, VKB_Query
- EMR Database access is required during rule execution
  - Presence_criterion,
    adverse_reaction_presence_criterion,
    observation_presence_criterion,
    medication_presence_criterion,
    comparison_criterion, VMR_query
How to translate

- **N-ary criterion**
  - Expression of BOOLEAN combination (AND, OR, or NOT) of simpler criterion expression
  - Each expression is mapped to one rule expression and connected with logical operator
  - Connected expression is another rule expression

- **Variable_Comparison_Criterion**
  - compares the value of a variable to some other value.
  - Rule expression compare the value to element of interface XML
  - The value of ‘References As’ slot is translated into the element of interface XML
  - Interface XML is already made at the invocation time of CDSS service
How to translate

❖ **Presence** **-** **Criterion**
  - checks for presence or absence of coded concept in instances of a VMR class within the valid time
  - Translate the rule to check the value availability in interface XML
  - interfaceXML contains the data queried from EMR by ExecuteVMRQuery()

❖ **Comparison** **-** **Criterion**
  - Check for equality of data stored in EMR and variable or value
  - Translate the rule to compare the value in interface XML with defined operator
How to Translate

- N-ary criterion

<table>
<thead>
<tr>
<th>Comment</th>
<th>diastolic BP for Stage I with DM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RuleExprObject</td>
<td>SAGE</td>
</tr>
<tr>
<td>RuleName</td>
<td>HTMgtV1_3_evaluation_Instance_72</td>
</tr>
<tr>
<td>RuleText</td>
<td>Boolean HTMgtV1_3_evaluation_Instance_720 (IF (FIRE(&quot;HTMgtV1_3_evaluation_Instance_73&quot;)) and FIRE(&quot;HTMgtV1_3_evaluation_Instance_79&quot;)) THEN (true) ELSE (false) RESULTINFO (BOTH:&quot;&quot;))</td>
</tr>
</tbody>
</table>
How to Translate

- **Variable_Comparison_Criterion,**
How to translate

Workflow to translate
- Verify the guideline in SAGE according to SWM
- Identify the logical error
- Translate into u-BRAIN representation model
- Viewing the translated representation model
- Simulating the guideline
Implementation and Case study

- Pulgin Module

- Several Options
Implementation and Case study

Verification Report
Implementation and Case study

📍 Translated Guideline
Implementation and Case study

Translated Results
Implementation and Case study

❖ Translated Results

Criterion 2

<table>
<thead>
<tr>
<th>Label</th>
<th>Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Today BP state is controlled without DM</td>
<td>diastolic BP for Controlled state without DM, systolic BP for Controlled state without DM</td>
</tr>
</tbody>
</table>

Converted to

Rule

```
{} Comment
{} RuleExprObject... SAGE
{} RuleName
{} RuleText
```

DIA Query

```
Comparison_Criterion_Query = query_id = HTMgV1_3_evaluation_Instance_129 = evmr_class = Patient = agg_operator = most_recent = Text = Select EMR_PATIENT.DBP, EMR_PATIENT.VS_DATE from EMR_PATIENT where EMR_PATIENT.ID = Datainput.pid
```
# Evaluation in Lab alerting CDSS

- 10 kinds lab test

## Environment

<table>
<thead>
<tr>
<th>Env</th>
<th>Server</th>
<th>Test Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>1.86GHz</td>
<td></td>
</tr>
<tr>
<td>Memory</td>
<td>1.5GB</td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td>windows2003 SP1</td>
<td></td>
</tr>
</tbody>
</table>

## Performance

<table>
<thead>
<tr>
<th># of cases</th>
<th>Turnaround Time of DI</th>
<th>Turnaround Time of KE</th>
</tr>
</thead>
<tbody>
<tr>
<td>323,445</td>
<td>346.16</td>
<td>51.90</td>
</tr>
</tbody>
</table>

## Correctness

<table>
<thead>
<tr>
<th>item</th>
<th># of cases</th>
<th>Error ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIA</td>
<td>323,445</td>
<td>0%</td>
</tr>
<tr>
<td>Knowledge engine</td>
<td>323,445</td>
<td>0%</td>
</tr>
</tbody>
</table>
SAGE Guideline execution environment is available

In the future
- Several case studies is going now.
- Verification environment will be added
  - So far, debugging utility verify the SAGE model corresponding structured workflow model
  - We have a plan to develop verification tool based on test case
- Develop knowledge repository management tools
  - Access control
  - Version control
  - Change control
  - Configuration management
- Reuse
Thank You!

Executable Guideline