Adaptive Loose Coupling Intelligent Rule System (ALCIRS)

Thesis Group Meeting #4
Presented by: M.M. Irfan Subakti – 1054257
Supervisor: Prof. John A. Barnden
Thesis Group Members: Dr. Mark G. Lee (RSMG rep)
Dr. John A. Bullinaria

School of Computer Science
University of Birmingham
United Kingdom
30 July 2013
Outline

- Abstract
- Review of previous work
- RBS: the test case
- Some detail
- Work done so far
- Reference
Abstract

- Rule-Based Systems (RBS) → very active research area
  - ALCIRS: an enhancement of VCIRS
- Rule dependency in RBS → loose coupling approach
- Adaptability → adaptive rule approach
- Contextual ontology → intelligence’s aspect for dealing with context dependence
- Support loose coupling aspect & producing new rule
- A novel adaptive, loose coupling self-learning rule system
  - Test case: readability analysis (Razon, 2013)
Review of previous work

- VCIRS

Evolutionally performance improving Variable-Centered Intelligent Rule System (VCIRS)

Simplify knowledge building

Empower knowledge inferencing

Evolutionally performance improving

Thesis Group Report #4 | ALCIRS | Irfan Subakti
**RBS: the test case**

- RB for readability analysis (Razon, 2013)
  - The effect of various English language syntactic and semantic features on each school grade level of second language (L2) learners → readability analysis tool: context-based & grammar-related submodules
Some detail (1)

- **ALCIRS**
  - **Adaptability**
    - When the rules changed over time by different cases and contexts
    - The chance and alternatives for producing any new rules which fit with the existing ones
  - **Loose coupling**
    - The dependency of parts of rules in a particular situation
  - **Contextual ontology**
    - The ability of the system to learn for obtaining the context from the content of the information from the rules
    - Support the adaptability aspect
Some detail (2)

- **ALCIRS**
  - **Loose coupling**
    - Interesting in Razon’s system (Razon, 2013) \(\rightarrow\) when a system is able to use parts of rules in loose coupling fashion \(\rightarrow\) more alternatives to get the same/similar conclusion/the result
    - The dependencies among rules will be measured \(\rightarrow\) advise the user about some alternatives \(\rightarrow\) new/different perspective how the result/conclusion might come up, exploring new frontier
  - **Adaptability**
    - Interesting in following the time/context changes, whether:
      - Want to suit the result/conclusion (which is resulted by these changes), or
      - Want to fulfil those changed requirement (which is necessary to get the new perspective of the result/conclusion)
    - Presenting the existing rules, the alternatives there might be, the better result for this situation
    - Ability to produce new rules that fit with the existing ones \(\rightarrow\) rule learning for generating new rules \(\rightarrow\) new/different perspective other than the existing rules
  - **Contextual ontology**
    - Learning the existing rules, gaining the context from the content of the information from the rules \(\rightarrow\) put these context and context together into structure called contextual ontology
    - Help and guide the loose coupling & adaptability features for suiting the changes required for getting the given goal
Some detail (3)

- Benslimane et al. (2006): Some concept definitions in multiple contexts for contextual ontology

Example 1. In context $s_1$, an employee is defined as anyone who has an employee number and in context $s_2$ as anyone who works for accompany.

$$Employee = (\exists Employee.Number.Number)[s_1] \cup (\exists WorksFor.Company)[s_2]$$

Example 2. A student is defined as a person who is enrolled in at least one course in context $s_1$, and it is defined as a person who has an id-card in context $s_2$.

$$Student = Person \cap ((\exists EnrolledIn.Course)[s_1] \cup (\exists Has.StudentIDCard)[s_2])$$

Example 3. A married man is defined as a man who has exactly a wife in context $s_1$, while in context $s_2$ he may have up to 4 wives and in context $s_3$ he may have an unlimited number of wives.

$$MarriedMan = Man \cap \exists woman.Woman \cap ((\leq 1 \text{ wife})[s_1] \cup (\leq 4 \text{ wife})[s_2] \cup (\top)[s_3])$$

The expression $(\top)[s_3]$ means the whole domain $\Delta$ in $s_3$, which express the absence of number constraint on $\text{wife}$ in $s_3$. 

Thesis Group Report #4 | ALCIRS | Irfan Subakti

30 July 2013
Some detail (4)

- Saleem and Bellahsene (2008): architecture for tree mining ontology concepts and taxonomy
- Ontology building from hierarchical structure source
Some detail (5)

- Zhang et al. (2010): the workflow of automatic ontology building with case-base
- Ontology building from case base input
Some detail (6)

- Contextual ontology might be useful for readability analysis → it gives a path for dealing with the context-based features
  - E.g., concepts in that domain whose nature is context sensitive: number of sentences, number of words and syllables, number of lines for a good paragraph - since they depend on which topic has been chosen
- Rules might be adapted because:
  - An initial set of rules might not work very well & then area adapted to work better, or
  - New data came along that make makes adaptation desirable, or
  - Both of cases above
Three aspects (adaptability, loose coupling and contextual ontology) in ALCIRS can be investigated as part of evaluation → how well the adjustment process work for the occurrences of each aspect.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Adaptability</th>
<th>Loose coupling</th>
<th>Contextual ontology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Concept Indexing (CI) will be used in Razon’s system (Razon, 2013) for the context-based analysis submodule of a readability analysis tool.

- Capture the semantic information

CI is proposed by Karypis and Han (2000):

- A fast dimensionality reduction algorithm which is able to capture the semantic similarities of texts
- It calculates a $n$-dimensional representation of corpus (a collection of documents) by first clustering the documents into $n$ groups, and then using the centroid or mean vectors of the clusters to derive the axes of the reduced $n$-dimensional semantic space.
Some detail (9)

- **ALCIRS**
  - Inspired by our previous work (Subakti, 2005 & 2006) for generating new rules procedure, we developing the framework for generating new rules from the existing ones by taken into account the statistics taken in the rule building to get the relative order or any parts of the rules combined by the idea of CI.
  
  - In the case of the presence of the terms in the various centroids, one observation can be seen is that the prevalent terms of the various centroids often contain terms which act as synonyms within the context of the topic they describe (Karypis and Han 2000)
Work done so far

- Java Architecture for XML Binding (JAXB) 2.1.10 has been implemented for XML programming and for rule formatting
- WordNet 2.1 (Windows version)
- Jawa WordNet Library (JWNL) 1.4 RC 3
- ConceptNet 5.1
- ALCIRS is remained to be developed but I/O, windows programming, etc. are already done in VCIRS. We will modify the implemented stuffs in VCIRS to be used in ALCIRS.
Future work

• Continue reading the literatures related
• Continue exploring & exploiting the issues came up
Reference


