Ontology-based intelligent decision support agent for CMMI project monitoring and control

Chang-Shing Lee *, Mei-Hui Wang, Jui-Jen Chen

Department of Computer Science and Information Engineering, National University of Tainan, Tainan 700, Taiwan

Accepted 14 June 2007
Available online 4 July 2007

Abstract

This paper presents an ontology-based intelligent decision support agent (OIDSA) to apply to project monitoring and control of capability maturity model integration (CMMI). The OIDSA is composed of a natural language processing agent, a fuzzy inference agent, and a performance decision support agent. All the needed information of the OIDSA, including the CMMI ontology and the project personal ontology, is stored in an ontology repository. In addition, the natural language processing agent, based on the Chinese Dictionary, periodically collects the information of the project progress from project members to analyze the features of the Chinese terms for semantic concept clustering. Next, the fuzzy inference agent computes the similarity of the planned progress report and actual progress report, based on the CMMI ontology, the project personal ontology, and natural language processing results. Finally, the performance decision support agent measures the completed percentage of the progress for each project member. The results provided by the OIDSA are sent to the project manager for evaluating the performance of each project member. The experimental results show that the OIDSA can work effectively for project monitoring and control of CMMI.

© 2007 Elsevier Inc. All rights reserved.

Keywords: Ontology; Intelligent decision support agent; CMMI; Fuzzy inference

1. Introduction

An ontology is a collection of key concepts and their inter-relationships collectively providing an abstract view of an application domain [4]. With the support of the ontology, both user and system can communicate with each other by the shared and common understanding of a domain [19]. In addition, an ontology is an essential element in many applications, including agent systems, knowledge management systems, and e-commerce platforms. For example, Corby et al. [1] proposed an ontology-based search engine to handle Resource Description Framework (RDF) Schema, part of Web Ontology Language (OWL) Lite, and RDF metadata. Huang and Murphey [20] presented a text document categorization system and applied text-mining technology to the automatic mapping of problem descriptions to correct diagnostic categories. Guarino et al. [2]...
developed OntoSeek, an information-retrieval system, to target online yellow pages and product catalogs, and examine linguistic ontologies’ role in content matching. Tu et al. [3] proposed an ontology-based configuration of problem-solving methods and generation of knowledge-acquisition tools to protocol-based decision support. Francisco et al. [16] developed an ontology-based intelligent web portal system to serve as service provider in recruitment tasks. In addition, Lee et al. [4] presented a fuzzy ontology and applied it to news summarization. They also proposed [17] a novel episode-based ontology construction mechanism to extract domain ontology from unstructured text documents.

An agent is a physical or virtual entity that is capable of acting in an environment and communicating directly with other agents [4]. Many applications using decision support agents have been proposed. For example, Lee et al. proposed a fuzzy decision agent for meeting scheduling support system [5], and a genetic fuzzy agent for meeting scheduling system [6]. Hamdi [7] developed a multi-agent information customization system that adopts the machine-learning paradigm to advise students by mining the Web. Delen and Pratt [8] designed and developed an intelligent decision support systems for manufacturing systems. Yan et al. [9] developed a multi-layer perceptron-based medical decision support system to support the diagnosis of heart diseases.

Capability maturity model integration (CMMI) is a model for process improvement and provides an opportunity to avoid or eliminate the bottlenecks and barriers that exist in organizations through integrated models that transcend disciplines [10]. There has been considerable research on CMM/CMMI: Yoo et al. [11] proposed a unified model for International Organization for Standardization (ISO)-certified organizations to implement CMMI, Huang and Han [12] presented a decision support model to assist managers in determining the priorities of the CMMI process areas, and Ronchetti et al. [13] presented an early estimation of software size in object-oriented environments developed by a CMM level 3 software firm. Niazi et al. [18] focused on software process improvement and designed three individual components to assist software process improvement. In this paper, we present an ontology-based intelligent decision support agent (OIDSA) to apply to project monitoring and control of CMMI. The proposed OIDSA is composed of a natural language processing agent, a fuzzy inference agent, and a performance decision support agent, to carry out the completed percentage of the project progress of each project member, and then send the results to the project manager for evaluating the performance of each project member.

This paper is organized as follows. Section 2 presents the structure of CMMI ontology. In Section 3, the ontology-based intelligent decision support agent is introduced. A fuzzy inference agent for performance decision support agent is proposed in Section 4. The experimental results are shown in Section 5. Finally, some conclusions are drawn in Section 6.

2. The structure of CMMI ontology

The purpose of the work presented in this article is the development of an OIDSA agent, which is based on an ontology model, for evaluating the completed percentage of progress, and providing appropriate information for project manager as a basis of assessing project members’ performance. Moreover, owing to the fact that software is increasingly becoming a larger part of many products and services, and the quality of a system is highly influenced by the quality of the process, a well-defined software development process plays a big part in the quality of a system. A CMMI is a reference model of mature practices in a specified discipline used to improve and appraise a group’s capability to perform that discipline. In addition, a CMMI model provides guidance when developing or improving the organization’s processes, and the ability to manage the development, acquisition, and maintenance of products or services. There are two representations, continuous and staged, and the components of both representations are process areas, specific goals, specific practices, generic goals, generic practices, typical work products, sub-practices, notes, discipline amplifications, generic practice elaborations, and references [10].

The maturity level 2 process areas of CMMI are requirement management, project planning, project monitoring and control, supplier agreement management, measurement and analysis, process and product quality assurance, and configuration management. The project monitoring and control process area of CMMI contains two specific goals. Each specific goal has some specific practices to achieve the associated specific goal. Specific goal 1 is to monitor if actual performance and progress of the project against the project plan. Specific
Table 1
Practice to goal relationship of project monitoring and control process area

<table>
<thead>
<tr>
<th>Specific goal</th>
<th>Specific practice</th>
</tr>
</thead>
</table>
| Monitor project against plan | 1. Monitor project planning parameters
| Monitor commitments | 1.2. Monitor commitments
| Monitor project risks | 1.3. Monitor project risks
| Monitor data management | 1.4. Monitor data management
| Monitor stakeholder involvement | 1.5. Monitor stakeholder involvement
| Conduct progress reviews | 1.6. Conduct progress reviews
| Conduct milestone reviews | 1.7. Conduct milestone reviews

Table 1 shows the specific practice to specific goal relationship of project monitoring and control process area [10].

The purpose of project monitoring and control is to provide an understanding of the project’s progress so that appropriate corrective actions can be taken when the project’s performance deviates significantly from the plan. Goal 2 is to manage corrective action to closure when the project’s performance or results deviate significantly from the plan. Table 1 shows the specific practice to specific goal relationship of project monitoring and control process area [10].

The purpose of project monitoring and control is to provide an understanding of the project’s progress so that appropriate corrective actions can be taken when the project’s performance deviates significantly from the plan. However, a project’s documented plan is the basis for monitoring activities, communicating status, and taking corrective action. Progress is primarily determined by comparing actual work product and task attributes, effort, cost and schedule to the plan at prescribed milestones or control levels within the project schedule or work breakdown structure. Also, because progress monitoring typically includes periodically measuring the actual completion of activities and milestones, comparing actual completion of activities and milestones against the schedule documented in the project plan, and identifying significant deviations from the schedule estimates in the project plan [10], project members must periodically fill in progress reports. Meanwhile the project manager must regularly review each project member’s progress, performance, and results, to identify and document significant issues and deviations from the plan. Based on the planned progress report for subsequent work and actual progress report for this work, the OIDSA is able to evaluate the completed percentage of progress, which not only reduces the cost of the project and effort of humans, but also provides a basis for project manager to evaluate the performance of each project member.

Based on the fundamental CMMI knowledge described above, the Chinese ontology for requirement management, project planning, and project monitoring and control process area of CMMI has been presented in this paper. The structure of the ontology, including the domain layer, category layer, and concept layer [4], is shown in Fig. 1. But for the ontology and all of reports, they are in Chinese but they have been translated into English for this paper. The concept layer is divided into five sub-layers, namely who layer, when layer, what layer, where layer, and how layer. In the domain layer, the domain name of this ontology is “Partial CMMI Level 2 Process Areas.” In the category layer, the requirement management, project planning, and project monitoring and control are the categories of the ontology. Each concept in the concept layer contains a concept name with several attributes and operations. For example, there are three concepts, “Life cycle,” “Schedule,” and “Milestone” in the when layer. The concept “Life cycle” contains two attributes, namely “Project life cycle” and “Project life cycle phases.” And its operations are “Planning,” “Scope,” “Assess,” and “Decide.” Take for another example. There are three concepts, “Manager,” “Project Staff,” and “Person” in who layer. The concept “Project Staff” contains nine attributes, namely “Stakeholder,” “Critical Roles,” “Customer,” “Staff,” “Relevant Stakeholder,” “Project Staff,” “Development Group,” “Supplier,” and “Requirement Provider.” The operations are “Join,” “Interact,” “Record,” “Review,” and “Provide.”

3. Ontology-based intelligent decision support agent

In this section, we utilize the CMMI ontology and project personal ontology to help perform an ontology-based intelligent decision support agent (OIDSA). Fig. 2 shows the structure of the OIDSA, including the natural language processing agent, the fuzzy inference agent, and the performance decision support agent.

The CMMI ontology and project personal ontology are pre-defined by CMMI experts and project domain experts, respectively. The project members periodically fill in the planned and actual progress reports and then store them in the project progress repository. Based on the Chinese Dictionary developed by the Chinese knowledge and information processing group [4], the natural language processing agent collects the Chinese
Fig. 1. Structure of the CMMI ontology for requirement management, project planning, and project monitoring and control process areas.
progress reports from the project progress repository. Then, the document pre-processing mechanism tags the terms of progress reports with Part-of-Speech (POS) such as noun, verb, and adjective. The term filter keeps the meaningful terms whose POS is noun or verb, and then passes these filtered terms to the fuzzy inference agent. The fuzzy inference agent makes use of the meaningful term sets of the planned term set and the actual term set, to infer the membership degrees belonging to the CMMI ontology. According to the membership degrees for planned term set and actual term set, the performance decision support agent is able to measure the progress as a completed percentage of project activities. In addition, the project manager regularly reviews the completed progress percentage of project members to identify and document if there are significant issues and deviations from the plan, while evaluating the performance of project members. Finally, the project domain expert retrieves the work breakdown structure of the project planning to construct and modify the project personal ontology.

Next, the natural language processing agent is introduced [17]. The fuzzy inference agent will be further described in Section 4. The three factors, including POS similarity, Number similarity, and Distance similarity, were selected as the conceptual similarity factors for making analysis of the Chinese terms and calculating the conceptual similarity between any two Chinese terms based on the features of the Chinese language and the definitions of the Chinese knowledge and information processing group. The POS similarity represents the path length between two nodes located on the tagging tree, shown in Fig. 3. The tagging tree is adopted to calculate the conceptual similarity in POS between any two Chinese terms. Besides, the POS path is bounded in the interval [0,6] according to the definitions of the Chinese knowledge and information processing group. For example, if there is a term pair (Project Progress, Monitor) with the POS (Na, VC), then the value for POS similarity of this term pair is 4 (Na → N → NV → V → VC), where Na, N, NV, V, and VC represent common noun, noun, noun-verb, verb and transitive verb, respectively.

The Number similarity represents the value of the conceptual similarity between any two Chinese terms according to these three Chinese characteristics: (1) the more identical words in both terms in the pair, the more similar the terms are to each other in semantic meaning; (2) terms in a pair with both identical and continuous words have much greater semantic similarity than those in a pair without identical or continuous words, and (3) terms in a pair with identical starting or ending words have a strong semantic similarity [17]. For example, the term pair (Project Staff, Project Manager), whose Chinese term pair is (專案人員, 專案經理), has two identical Chinese words, “專” and “案”, and the identical starting Chinese word, “專”, so the value of Number similarity of this term pair is 2.5, i.e., 2 plus 0.5 equals 2.5.
The Distance similarity represents the semantic distance in the same layer of domain ontology for any term pair. In the Distance similarity computation, domain experts pre-define the concept layer of CMMI ontology as five sub-layers, including the who layer, the when layer, the what layer, the where layer, and the how layer. Take this term pair (Life cycle, Milestone), for example. These two terms are both located in the when layer of Fig. 1. Hence, the Distance similarity is 2 between these two terms, i.e. Life Cycle → Schedule → Milestone.

4. Fuzzy inference agent for performance decision support mechanism

In this section, the fuzzy inference agent for performance decision support mechanism, including fuzzy inference agent and performance decision support agent, is introduced. The following two sub-sections describe the fuzzy inference agent and the performance decision support agent for obtaining the completed percentage of progress, respectively.

4.1. Fuzzy inference agent

The architecture of fuzzy inference agent in the OIDSA, including the input linguistic layer, input term layer, rule layer, output term layer, and output linguistic layer, is shown in Fig. 4. The inputs of input linguistic layer include the Chinese term set of planned progress report, actual progress report, and CMMI ontology. The nodes in this layer directly transmit the values of POS similarity, Number similarity, and Distance similarity for the term sets to the next layer.

The second layer, input term layer, performs the membership functions to compute the membership degrees for all terms derived from the retrieved planned term set, actual term set, and all concepts of CMMI ontology. There are three fuzzy variables, including POS similarity, Number similarity, and Distance similarity, considered in this layer for each term’s property. Fig. 5 shows the trapezoidal membership function \((x: a, b, c, d)\) for fuzzy set \(A\) [21], and the trapezoidal membership function denoted by Eq. (1) is represented as \([a, b, c, d]\) in this paper.

\[
\text{trapezoid}(x : a, b, c, d) = \begin{cases} 
0, & x < a \\
(x - a)/(b - a), & a \leq x < b \\
1, & b \leq x < c \\
(d - x)/(d - c), & c \leq x < d \\
0, & x \geq d
\end{cases}
\] (1)
For fuzzy variable POS, there are three linguistic terms, POS_Low, POS_Medium, and POS_High, to express concepts and knowledge of POS, whose trapezoidal membership functions are represented as [4, 6, 6, 6], [1, 3, 3, 5], and [0, 0, 0, 2], respectively. Fig. 6a shows the membership functions for fuzzy variable POS. If the POS value for a term pair is small, then the membership degree for the POS similarity is high. For example, if a term pair has the same POS, then the POS value for this term pair is 0. The membership degree for the POS similarity is high, i.e., the membership degree is 1 in this case. Therefore, if the POS value for a term pair is large, then the membership degree for the POS similarity is low. The Number fuzzy variable defines three linguistic terms, namely Number_Low, Number_Medium, and Number_High, whose trapezoidal membership functions are denoted as [0, 0, 0, 0.3], [0.2, 0.5, 0.5, 0.8], and [0.7, 1, 1, 1], respectively. Fig. 6b shows the membership functions for fuzzy variable Number. If the Number value for a term pair is small, then the
membership degree for the *Number* similarity is low. On the contrary, if the *Number* value for a term pair is large, then the membership degree for the *Number* similarity is high. The linguistic terms of fuzzy variable *Distance* are *Distance_Low*, *Distance_Medium*, and *Distance_High*, whose trapezoidal membership functions are represented as [11, 16, 16, 16], [4, 8, 8, 12], and [0, 0, 0, 5], respectively. **Fig. 6c** shows the membership functions for fuzzy variable *Distance*. The *Distance* value represents the semantic distance of a term pair. If the *Distance* value for a term pair is small, then the membership degree for the *Distance* similarity is high. Take a term pair with the same terms as an example. The *Distance* value for this term pair is 0. In this case, the membership degree for the *Distance* similarity is high, i.e., the membership degree is 1. In other words, if the *Distance* value for a term pair is large, then the membership degree of the *Distance* similarity is low. In **Fig. 4**, each node located in the third layer, *rule layer*, represents a fuzzy rule. The main task of this layer is to be responsible for using AND operator to combine the matching degree of each fuzzy rule’s condition. Herein, the MIN operator is adopted as the fuzzy conjunction operator. In our model, the rules are defined by domain expert’s knowledge previously, and we show them in Table 2. The node located in the *output term layer*, the fourth
layer, performs the consequent of the fuzzy rules. We utilize Strength fuzzy variable with five linguistic terms in this layer. The linguistic terms are Strength_VeryLow, Strength_Low, Strength_Medium, Strength_High, and Strength_VeryHigh, whose trapezoidal membership functions are denoted as [0, 0, 0, 0.3], [0.2, 0.3, 0.3, 0.5], [0.3, 0.5, 0.7, 1], [0.5, 0.7, 0.8, 0.8], and [0.7, 1, 1, 1], respectively. Fig. 6d shows the membership functions for fuzzy variable Strength. If the Strength value for a term pair is large, then the membership degree for the Strength similarity is high. On the contrary, if the Strength value for a term pair is small, then the membership degree for the Strength similarity is low.

Fig. 7 shows the structure of the node in the output term layer for the fuzzy rules triggering the fuzzy set Strength_Low. Take the Rule2, Rule3, Rule4, Rule5, Rule7, Rule10, Rule11, Rule13, and Rule19, as an example. They all trigger the same fuzzy set, Strength_Low. The fuzzy inference agent performs the MAX operation to integrate the triggered rules and outputs the maximum center of area [4] for fuzzy set Strength_Low, to the output linguistic layer. The node on the output linguistic layer, the final layer, performs the final Strength value using the ordered weighted averaging aggregation operator [14,15].

4.2. Performance decision support agent

The performance decision support agent is responsible for measuring the completed percentage of project progress based on the output of the fuzzy inference agent. The smaller the difference in evaluated percentage between project members and the OIDSA is, the better the performance of the OIDSA is. The following algorithm shows the process of the performance decision support agent.

**Algorithm for performance decision support agent**

**Input:**
1. All concepts \((C_1, \ldots, C_m)\) of the CMMI ontology.
2. All terms \((PT_1, \ldots, PT_n)\) of the planned progress report.
3. All terms \((AT_1, \ldots, AT_r)\) of the actual progress report.

**Output:**
1. Matched concepts in the planned term set.
4. Estimated completed percentage of project progress.
Method:

Step1: For all concepts \((C_1, \ldots, C_m)\) of the CMMI ontology

Step1.1: For all terms \((PT_1, \ldots, PT_n)\) of the planned term set

Step1.1.1: Retrieve a pair \((C_u, PT_v)\).

Step1.1.2: Compute the values of \(POS\), \(Number\), and \(Distance\) for \((C_u, PT_v)\).

Step1.1.3: Generate the \(Strength\) value after performing the fuzzy inference agent.

Step1.1.4: If \(Strength\) value of \((C_u, PT_v)\) then

Step1.1.5: Obtain the matched concepts in the planned term set.

Step2: For all concepts \((C_1, \ldots, C_m)\) of the CMMI ontology

Step2.1: For all terms \((AT_1, \ldots, AT_r)\) of the actual term set

Step2.1.1: Retrieve a pair \((C_u, AT_w)\).

Step2.1.2: Compute the values of \(POS\), \(Number\), and \(Distance\) for \((C_u, AT_w)\).

Step2.1.3: Generate the \(Strength\) value after performing the fuzzy inference agent.

Step2.1.4: If the \(Strength\) value of \((C_u, AT_w)\) then

Step2.1.5: Obtain the matched concepts of the actual term set.

Step3: For \(i = 1\) to \(p\) /*The \(p\) denotes the number of matched concepts in the planned term set.*/

Step3.1: For \(j = 1\) to \(q\) /*The \(q\) denotes the number of matched concepts in the actual term set.*/

Step3.1.1: If \((C_i == C_j)\) then

Step3.1.1.1: \(k = k + 1\) /*The \(k\) denotes the number of matched concepts in the planned term set and the actual term set simultaneously.*/

Step4: Compute the completed percentage of project progress: \(\frac{k}{p} \times 100\%\)

Step5: End.

5. Experimental results

We have constructed an experimental platform at National University of Tainan to test the performance of the proposed approach. The experimental Chinese progress reports are retrieved from three project members involved in a CMMI project at National University of Tainan. Every project member reported his planned and actual progress each week. The first experiment is to evaluate the performance of the OIDSA under various \(\sigma_{Strength}\) values. The \(\sigma_{Strength}\) value denotes a threshold for testing the effect of membership degree for the \(Strength\) fuzzy variable. Fig. 8a–d shows the curves of the completed percentage of progress evaluated by the OIDSA and project members, where \(\sigma_{Strength}\) denotes 0.5, 0.75, 0.8, and 0.9, respectively. Table 3 lists the values of the completed percentage of progress from the OIDSA and project members under the above-mentioned various \(\sigma_{Strength}\) values. We observe that when \(\sigma_{Strength}\) is 0.5, the completed percentage of progress
evaluated by the OIDSA is 100% for all progress reports, which significantly deviated from the estimation of the project members, except for the 9th progress report. But from Fig. 8c, we also can observe that the OIDSA can obtain the correct results for most progress reports when \( \sigma_{\text{Strength}} \) is 0.8. In this case, the OIDSA is capable of correctly estimating the completed percentage of the progress for the progress reports to meet the project members’ evaluation. Therefore, choosing a suitable \( \sigma_{\text{Strength}} \) value is very important for the OIDSA.

Table 3
The values of the completed percentage of the progress evaluated by the OIDSA with various \( \sigma_{\text{Strength}} \), and by project members

<table>
<thead>
<tr>
<th>Progress report no</th>
<th>Estimated percentage of progress (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OIDSA ( \sigma_{\text{Strength}} )</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
</tr>
<tr>
<td>11</td>
<td>100</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>13</td>
<td>100</td>
</tr>
<tr>
<td>14</td>
<td>100</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
</tr>
</tbody>
</table>

Fig. 8. The curves of completed percentage of the project progress evaluated by the OIDSA and project members, when \( \sigma_{\text{Strength}} \) is: (a) 0.5, (b) 0.75, (c) 0.8, and (d) 0.9.
The second experiment is to observe the behavior of the OIDSA from the performances of the mean absolute error value. Fig. 9 shows the curve of the average value of mean absolute error for all reports under various $\sigma_{strength}$. Table 4 lists the values of average value of mean absolute error for all reports under various $\sigma_{strength}$. In addition, Fig. 10 also shows the curves of the value of mean absolute error for each progress report under various $\sigma_{strength}$. From the results shown in Fig. 9, Table 4, and Fig. 10, we also observe that the worst and best case occurs, when $\sigma_{strength}$ is set to 0.5 and 0.8, respectively.

In the final experiment, Fig. 11a–d displays the bar charts of the completed percentage of progress evaluated by the project member and by the OIDSA under various $\sigma_{strength}$ values for the 1–4th, 5–8th, 9–12th, and 13–15th progress reports, respectively. From Fig. 11, we also know that $\sigma_{strength}$ would affect the correctness of the OIDSA, and that setting $\sigma_{strength}$ to 0.8 can acquire the best results. In a word, the experimental results show that the performance of the OIDSA is deeply affected by the value of $\sigma_{strength}$. Table 5 shows the detailed experimental results of the 7th progress report under the condition setting $\sigma_{strength}$ to 0.8. In Table 5, it indicates that the number of matched concepts in the 7th planned term set is 7, namely “Life cycle,” “Project

![Fig. 9. The curve of the average value of mean absolute error for all reports under various $\sigma_{strength}$.](image)

<table>
<thead>
<tr>
<th>$\sigma_{strength}$</th>
<th>0.5</th>
<th>0.75</th>
<th>0.8</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average value of mean absolute error</td>
<td>34.13</td>
<td>10.78</td>
<td>4.66</td>
<td>4.93</td>
</tr>
</tbody>
</table>

![Fig. 10. The curves of the value of mean absolute error for each progress report under various $\sigma_{strength}$.](image)
The 7th planned progress report

Next week is planned to estimate of planning parameters. The planned practices are as follows
1. Consider project requirements, including the product requirements, the requirements imposed by the organization, the requirements imposed by the customer
2. Consider the scope of the project
3. Consider the identified tasks and work products
4. Consider technical approach
5. Consider selected project life-cycle model
6. Consider the attributes of the work products and tasks
7. Consider the schedule
8. Consider models or historical data for converting the attributes of the work products and tasks into labor hours and cost

Matched concepts in the 7th planned term set
Life cycle, Project planning, Schedule, Understand requirements, Budget, Estimates, Work package

The 7th actual progress report

The actual progress of this week is as below
1. Complete project requirements, including the product requirements, the requirements imposed by the organization, the requirements imposed by the customer
2. Complete identifying the scope of the project
3. Complete considering the attributes of the work products and tasks
4. Complete selecting the project life-cycle model

Matched concepts in the 7th actual term set
Life cycle, Project planning, Understand requirements, Work package

Matched concepts in the 7th planned term set and the 7th actual term set simultaneously
Life cycle, Project planning, Understand requirements, Work package

Completed percentage of progress evaluated by the OIDSA = 57%

---

Table 5
The 7th planned and actual progress reports

<table>
<thead>
<tr>
<th>Table 5 The 7th planned progress report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next week is planned to estimate of planning parameters. The planned practices are as follows</td>
</tr>
<tr>
<td>1. Consider project requirements, including the product requirements, the requirements imposed by the organization, the requirements imposed by the customer</td>
</tr>
<tr>
<td>2. Consider the scope of the project</td>
</tr>
<tr>
<td>3. Consider the identified tasks and work products</td>
</tr>
<tr>
<td>4. Consider technical approach</td>
</tr>
<tr>
<td>5. Consider selected project life-cycle model</td>
</tr>
<tr>
<td>6. Consider the attributes of the work products and tasks</td>
</tr>
<tr>
<td>7. Consider the schedule</td>
</tr>
<tr>
<td>8. Consider models or historical data for converting the attributes of the work products and tasks into labor hours and cost</td>
</tr>
</tbody>
</table>

Matched concepts in the 7th planned term set
Life cycle, Project planning, Schedule, Understand requirements, Budget, Estimates, Work package

<table>
<thead>
<tr>
<th>Table 5 The 7th actual progress report</th>
</tr>
</thead>
<tbody>
<tr>
<td>The actual progress of this week is as below</td>
</tr>
<tr>
<td>1. Complete project requirements, including the product requirements, the requirements imposed by the organization, the requirements imposed by the customer</td>
</tr>
<tr>
<td>2. Complete identifying the scope of the project</td>
</tr>
<tr>
<td>3. Complete considering the attributes of the work products and tasks</td>
</tr>
<tr>
<td>4. Complete selecting the project life-cycle model</td>
</tr>
</tbody>
</table>

Matched concepts in the 7th actual term set
Life cycle, Project planning, Understand requirements, Work package

Matched concepts in the 7th planned term set and the 7th actual term set simultaneously
Life cycle, Project planning, Understand requirements, Work package

Completed percentage of progress evaluated by the OIDSA = 57%
planning,” “Schedule,” “Understand requirements,” “Budget,” “Estimates,” and “Work package,” and that the number of concepts simultaneously matched in the 7th planned term set and the 7th actual term set is 4, namely “Life cycle,” “Project planning,” “Understand requirements,” and “Work package.” Therefore, we can acquire that the completed percentage of the progress evaluated by the OIDSA is $\frac{4}{7} = 57\%$.

6. Conclusions

In this paper, an ontology-based intelligent decision support agent (OIDSA) for CMMI project monitoring and control is proposed. The OIDSA contains three subagents, including the natural language processing agent, the fuzzy inference agent and the performance decision support agent. Besides, the CMMI ontology and project personal ontology are also presented in this paper. Furthermore, we also have constructed an experimental platform to test the proposed approach. The experimental results show that the OIDSA for CMMI project monitoring and control can effectively evaluate the completed percentage of progress to reduce the human efforts and the costs of the project. Besides, there are still some problems needed to further study in the future. For example, if the project member reports “To obtain an understanding of requirements” in the planned progress report, but reports “Failure in understanding of requirements” in the actual progress report, then such collisions are still resolved by the project manager. In addition, adding the learning mechanism to the fuzzy inference rules, providing Web service for users, and improving the precision of the proposed method are also our future tasks.

Acknowledgements

The authors thank the anonymous referees for their constructive and useful comments, and also thank the partially support sponsored by the Department of Industrial Technology, Ministry of Economic Affairs, ROC, under the Grants 96-EC-17-A-02-S1-029 and the National Science Council of Taiwan under the Grant NSC 95-2221-E-024-009-MY2.

References