

A Semantic-based Approach for Representing Successful Graduate Predictive Rules

Noppamas Pukkhem

Department of Computer and Information Technology
Faculty of Science, Thaksin University, Phattalung, Thailand 93110
noppamas@tsu.ac.th

Abstract— This paper seeks to identify the factors of university students in major of Computer Science at Thaksin University, Thailand that predicts successful completion of the bachelor's degree. Decision tree C4.5/J48, ID3 and ADTree algorithm, the classification algorithms in data mining which are commonly used in many areas can also be implemented to generate the classification rules. In our experiment with 128 training records, we found an overall accuracy of C4.5/J48 algorithm was 90.625%, ID3 algorithm and ADTree were 96.875%. Moreover, we extend the classification rule by applying a semantic-based approach for creating a classification tree ontology. The ontology represent about the classification rules that used to enable machines to interpret and identify learner factors in process of prediction. We also explain how ontological representation plays a role in classifying students to predictive target class. The inference layer of classification tree ontology is based on SWRL (Semantic Web Rule Language), making a clarify separation of the program component and connected explicit modules. One of the major advantages of the proposed approach is that identifying success factors will give students an awareness of essential features for successful completion of their graduate studies.

Keywords— Decision Tree, Graduate Prediction, Ontology, Semantic Web, SWRL

I. INTRODUCTION

Thaksin University, a university in a southern of Thailand, admits students under various programs from different educational backgrounds, school, place and various educational metric in entrance tests. Every year, The number of unsuccessful students or student who not completed their degree in 4 years in bachelor degree has increased. The survey reports about 10-15% from academic year 2011 to 2012. Since our work is that identifying success factors will give students an awareness of essential features for successful completion of their graduate studies. In this work, Classification data mining techniques with ID3, C4.5 and ADTree are used to generate the predictive model.

Semantic web, as a technological foundation for the next generation of web-based systems [1]. The main focus of semantic web is to support distributed information with well-defined meaning, understandable for humans as well as machines. Ontologies, the generally defined as a representation of a shared conceptualization of a specific

domain, is a main part of the semantic web. Recently many works have been applied semantic web for several aspects of their domain, such as task ontology, user model ontology, domain ontology and others [2].

In our work, we applied the semantic web technologies to represent about the classification rules that used to enable machines to interpret and identify learner factors in process of prediction.

The rest of the paper is organized as follows. In the section II gives background knowledge. An overview of classification algorithm and semantic-based tool is also included. Section III presents a proposed model that explaining a predictive model generation and representing into semantic-based rules. Finally, section IV concludes this paper, giving a summary of its main contribution and towards future research work.

II. BACKGROUND KNOWLEDGE

Data mining, the process of discovering knowledge in a large amount of data stored in databases or data warehouses, widely used in many research. These techniques are applied in various fields such as decision support system, machine learning, artificial intelligence and others. Our goal in this work is to predict the current student into 2 target classes-“Success” or “Fail”, that we have already known a supervised target class, so the classification algorithm will be used in predictive model generation. Then, the classification tree will be represented into ontology-based representation.

A. Classification Algorithm

1) ID3 algorithm: Iterative Dichotomiser 3 in an algorithm used to create a decision tree from the supervised training dataset based on information entropy[3, 8]. The information gain is a main criteria for feature selection; usually the feature that has a highest information gain will be selected to as the splitting feature at a current node, where gain is defined as the difference between how much information is needed after the split. The information gain $I(S)$ of the attribute is calculated by

$$I(S) = - \sum_{i=1}^m p_i \log_2(p_i)$$

where p_i is the probability that a arbitrary vector in S belongs to class. A logarithm function to the base 2 is used, because the information gain is encoded in bits. $I(S)$ is just

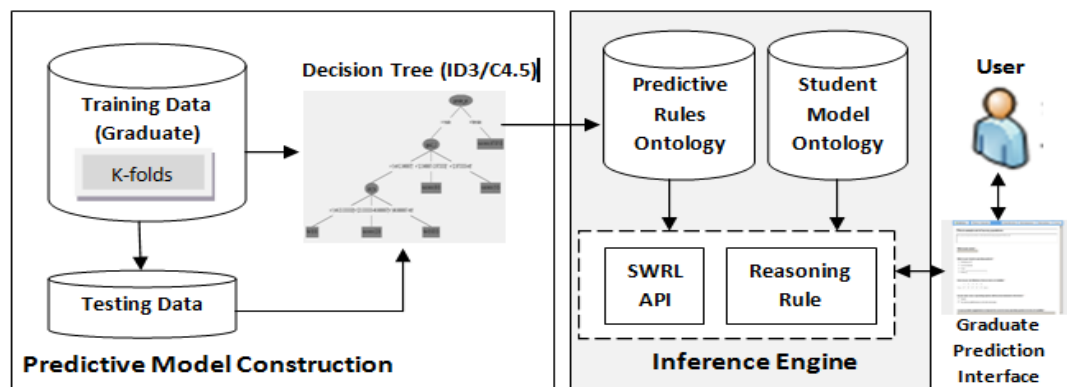


Figure 1. The overall achitecture of a semantic-based approach for representing predictive rules

the average amount of information needed to identify the class label in vector S .

2) C4.5 algorithm: C4.5 is an extended algorithm of the ID3 algorithm used to overcome its disadvantages such as, pruning the overfitting tree and handling attributes with continuous and discrete value[4, 7]. Moreover, C4.5 select one feature of the data that most effectively splits data set on each node of the tree.

3) ADTree algorithm: ADTree is a data mining boosting algorithm using typically used either decision tree as weak hypotheses [4]. Implementations are available in WEKA software. ADTree creates a set of T weighted decision trees where T is the number of boosting iterations. ADTree supports structure to the set of hypotheses by requiring that they build off a hypothesis that was produced in an earlier iteration.

B. SVM feature selection algorithm

SVM (Support Vector Machine) evaluates the feature based on the ranking of the features. It evaluates the worth of a features by using an SVM classifier [10]. Features are ranked by the square of the weight assigned by the SVM. For our experimentation we have used WEKA mining tool. All the classifiers and feature selection techniques are experimented using default parameters in WEKA.

C. Semantic-Based Tool

There are several semantic-based tools for information extraction and transform into meaningful which can be used through the process of building ontology-based model in our work:

1) OWL (Web Ontology Language): OWL is a language for definition of web ontologies which explicitly represents the semantics of terms in vocabularies and its relationships between these terms [1, 5]. This is an accepted standard, language and platform independent and well- formed XML-markup language.

2) RDF (Resource Definition Framework): RDF is a set of recommendations for well-formed XML documents. It is a more data aspect framework than human aspect. So, it is designed for enabling machine processing rather than for ease of human understanding.

3) SWRL (Semantic Web Rule Language): SWRL is a proposal in submission by the W3C, aiming at combining OWL and inference rules language [1, 2]. SWRL provides seven types of atoms: *Class*, *Individual*, *Data valued property*, *Different individual*, *Same individual*, *Built-in* and *Data range atoms*. A class atom consists of an OWL named class or class expression and a single argument representing an OWL individual. Examples include:

- $\text{Person}(?x)$
- $\text{Student}(?x)$
- $\text{Student}(\text{Botan})$

where Person and Student are OWL named classes, $?x$ is a variable representing and OWL individual, and Botan is the name of an OWL individual and refer to the name of the student.

An individual property atom consists of an OWL object property and two arguments representing OWL individuals. Examples include:

- $\text{hasGender}(?x, ?y)$
- $\text{hasSibling}(?x, ?z)$

where hasGender and hasSibling are OWL object properties that explain the relationship between two objects, $?x$ and $?y$ are variable representing OWL individual.

The idea behind the semantic-based representation is, as the name suggests, to add a level of meaning to the Web. As this reason, it can be more easily manipulated by computer programs, and more effectively by humans. The proposed ontology-based predictive model was presented in section III.

With this approach, supported our graduate predictive model higher scalability and easier maintenance of the approach are expected.

III. PROPOSED MODEL

A. Overall Architecture

The overall architecture, shown in Figure 1, has three main components. First, the *predictive model construction* that explains about a process of student classification into a target class. In order to generate a predictive model by using ID3, C4.5 and ADTree algorithm, a labelled class of student training data is used to construct a decision tree. In accuracy evaluation process, we spited 3-folds data set from a training data to be used as a testing data. Second, the *ontology-based inference engine* is presented. This component consists of predictive rule ontology, student model ontology and inference engine. The third component is the *graduate prediction interface* that supported the ease of use between the user and the predictive model.

B. Data Preprocessing and Predictive Model Construction

1) Training Dataset and Data Preprocessing:

For generation a classification model in our work, we prepared 128 training datasets consisting of information about students of Computer Science major who graduated in academic year 2010-2012. This data had details of each student such as full name, primary school, GAT score, GPA of each semester, the parent's background, etc. The number of features is 35 and stored in a training data database. We scan all dataset for managing the incomplete or inconsistence data. Next, the data will be encoded into defined feature's label. Data transformation process is required to transform continuous data into interval-data. Finally, we created the input of training dataset into ARFF file format and imported to WEKA. Figure 2 shows the snapshot of training data records in the WEKA viewer tool.

education_fa	old_ma	status_ma	salary_ma	career_ma	education_ma	status_fama	child_ed	status_cu	time
Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal
Primary	{inf-4...	yes	{inf-175...	contract	BeginSchool	cohabit	3	no	success
Primary	{47.6...	yes	{inf-175...	business	BeginSchool	cohabit	3	no	fail
BeginSchool	{56.3...	yes	{inf-175...	other	BeginSchool	separation	1	no	success
Primary	{inf-4...	yes	{inf-175...	other	LowPrimary	cohabit	3	no	fail
HighSchool	{inf-4...	yes	{inf-175...	business	HighVocational	cohabit	1	no	success
Vocational	{47.6...	yes	{inf-175...	other	LowPrimary	separation	2	no	success
HighVocational	{inf-4...	yes	{inf-175...	farmers	HighVocational	cohabit	1	no	success
Primary	{47.6...	yes	{inf-175...	farmers	LowPrimary	cohabit	3	no	success
Vocational	{47.6...	yes	{inf-175...	trade	Vocational	cohabit	2	no	success
Primary	{47.6...	yes	{inf-175...	trade	BeginSchool	cohabit	2	no	fail
Vocational	{inf-4...	yes	{inf-175...	other	HighSchool	cohabit	2	no	success
BeginSchool	{47.6...	yes	{inf-175...	other	LowPrimary	cohabit	3	no	success
HighSchool	{inf-4...	yes	{inf-175...	farmers	BeginSchool	cohabit	2	no	success
Primary	{inf-4...	yes	{inf-175...	farmers	LowPrimary	cohabit	2	no	fail
Primary	{47.6...	yes	{35000...	farmers	BeginSchool	cohabit	2	no	success

Figure 2. Example of training data records with WEKA Viwer.

2) Feature Selection:

In this process, we removed the irrelevant features from the dataset by using a GreedyStepwise feature selection, Ranker Method and SVM. The experimental results of selected features list comparing between 3 methods are shown in Table 1.

TABLE 1. SELECTED FEATURE LIST OF THREE FEATURE SELECTION METHODS

Methods	List of Selected Features
GreedyStepwise	[10 features] GAT_Score, Gender, GPA_1_1, GPA_2_1, GPA_2_2, Education_ma, Education_fa, Sibling, Disease, GPA_Sc
Ranker	[12 features] GAT_Score, Gender, GPA_2_2, GPA_2_1, GPA_1_1, GPA_1_2, GPA_Sc, Education_ma, Education_fa, Sibling, Disease, Salary_ma
SVM	[9 features] GAT_Score, Gender, GPA_2_2, GPA_2_1, GPA_Sc, Education_ma, Education_fa, Sibling, Disease

The comparison between the original feature set with all 35 features, and the reduced new feature set using proposed feature selection method is done.

For example the full name and student ID were irrelevant in predicting the future graduate, so it will be removed. The smallest feature set is made by SVM method and we will used a list of them to create a predictive model in next section. The list of selected features after using a SVM method are GAT_Score, Gender, GPA_2_2, GPA_2_1, GPA_Sc, Education_ma, Education_fa, Sibling and Disease. Then, the "target class" feature was added and it held the predicted result, which can be either "Success" of "Fail". Finally, all of selected features will be used to generate a classification model.

3) Model Construction and Evaluation

In model construction and evaluation, WEKA version 3.6.10 was used to demonstrate our predictive model. For example, we found the feature GAT_Score is most suited to be a root node of the decision tree.

By using the WEKA tool to demonstrate our work, the ID3 decision tree is shown in Figure 3. For evaluating the model effectiveness, we use a binary (positive and negative) classification. There are four possible outcomes of classifier prediction: True Negative (TN), Ture Positive (TP), False

Negative (FN) and False Positive (TP). A two-plan confusion matrix is explained in Table 2.

school), Sibling and Disease. All of feature will be used to create a semantic-based inference model.

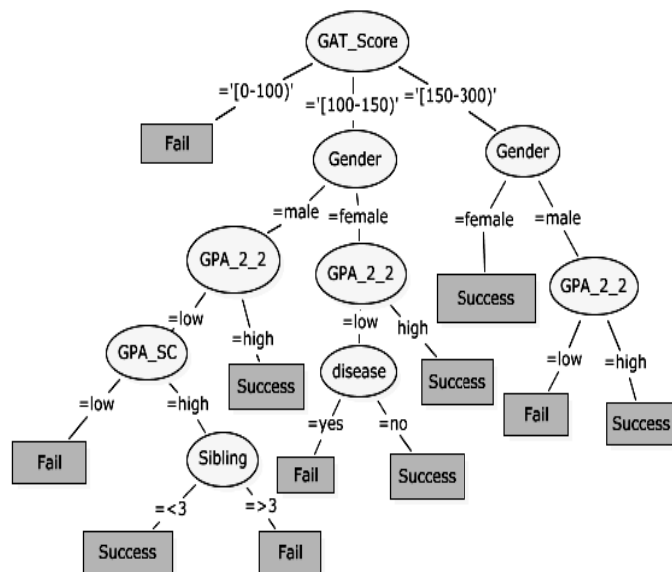


Figure 3. Example of Decision Tree of ID3 Algorithm

TABLE 2. CONFUSION MATRIX

Obtained Result	Correct Result	
	+(Success)	- (Fail)
+(Success)	TP	FP
- (Fail)	FN	TN

The four values TN, TP, FN and FP provided by the confusion matrix that are well known and typically used within the data mining, where N represents the number of instances in a given set. Mean Absolute Error (MAE) is the average of the difference between predicted and actual value in all test cases; it suggest that the error rate is very small, which can be considered as a measure of effectiveness of the model. It can be calculated by the following equation

$$MAE = \frac{|a_1 - c_1| + |a_2 - c_2| + |a_3 - c_3| + \dots + |a_n - c_n|}{N}$$

where N represents the number of records in a data set. This value is computed by taking the average of the absolute differences between each computed (c_i) and its corresponding correct value (a_i).

From an experiment, the result of accuracy and MAE are shown in Table3. In our experiment with 128 training records, we found an overall accuracy of C4.5/J48 algorithm was 90.625% , ID3 algorithm and ADTree was 96.875% but MAE(Mean Absolute Error) of ID3 is less than ADTree. The decision tree features of ID3 are GAT_Score, Gender, GPA_2_2(GPA of year 2, semester 2), GPA_Sc (GPA of high

TABLE 3. RESULTS OF CLASSIFICATION ALGORITHM COMPARISON

	Accuracy	MAE	Confusion Matrix
C4.5/J48	90.625%	0.1108	=== Confusion Matrix === a b <-- classified as 112 2 a = success 2 12 b = fail
ID3	96.875%	0.0313	=== Confusion Matrix === a b <-- classified as 111 3 a = success 9 5 b = fail
ADTree	96.875%	0.0877	=== Confusion Matrix === a b <-- classified as 114 0 a = success 4 10 b = fail

In order to develop the ontology-based prediction rules, example of an inductive rules as follows were prepared over the ID3 decision tree.

Rule#1: if $GAT_Score = "[0-100]"$ then $Class = "Fail"$

Rule#2: if $GAT_Score = "[100-150]"$
and $Gender = "male"$
and $GPA_2_2 = "low"$
and $GPA_SC = "low"$
then $Class = "Fail"$

Rule#3: if $GAT_Score = "[100-150]"$
and $Gender = "female"$
and $GPA_2_2 = "low"$
and $Disease = "yes"$
then $Class = "Fail"$

Rule#4: if $GAT_Score = "[100-150]"$
and $Gender = "female"$
and $GPA_2_2 = "high"$
then $Class = "Success"$

Rule#5: if $GAT_Score = "[100-150]"$
and $Gender = "male"$
and $GPA_2_2 = "high"$
and $GPA_SC = "high"$
and $Sibling = "<3"$
then $Class = "Success"$

Rule#6: if $GAT_Score = "[100-150]"$
and $Gender = "male"$
and $GPA_2_2 = "high"$
and $GPA_SC = "high"$

and Sibling = ">=3"
then Class="Fail"

Rule#7: if $GAT_Score = "[100-150]"$
and $Gender = "female"$
and $GPA_2_2 = "high"$
then $Class = "Success"$

Rule#8: if $GAT_Score = "[150-300]"$
and $Gender = "male"$
and $GPA_2_2 = "low"$
then $Class = "Fail"$

Rule#9: if $GAT_Score = "[150-300]"$
and $Gender = "male"$
and $GPA_2_2 = high$
then $Class = "Success"$

Then, we used the ontology approach to model the context of the decision tree rules and the student model. The explanation of our approach will be describe in next subsection.

C. Student Model Ontology

The student model is described by ontology-based for conceptualizing and exploited by the inference engine. For creating the student model ontology, we initial the process of classification that is explained in subsection B. Then, we adopt OWL (Web Ontology Language) to express ontology enabling expressive knowledge description and information interoperability of classes. According to the student model ontology, Figure 4 shows the part of the class hierarchical and the object properties that implemented in Protégé 4.3.0.[5]

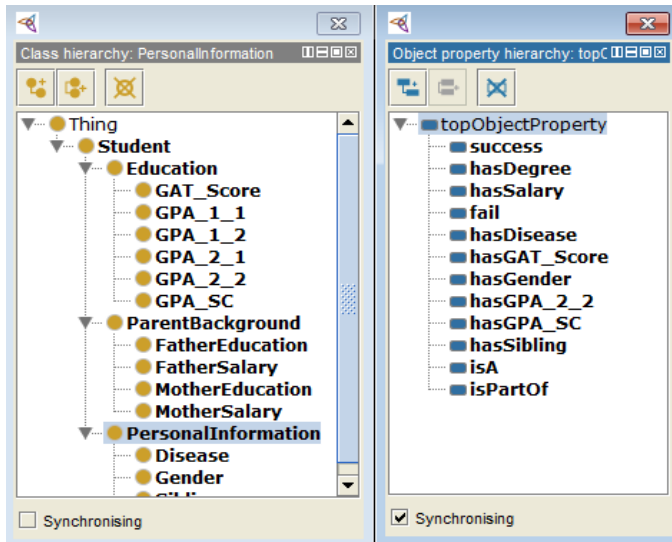


Figure 4. Example of Student Model Ontology

For example, the class student is explained by the class *PersonalInformation*. This class consists of three subclasses:

Gender, *Disease* and *Siblings*. The relation between the class can be described by the object properties.

D. Inference Method

The inferencing itself is a task of the used rule engine which can have different algorithm implemented for it. The inference method that decision tree algorithm are based on will be described next, namely successful graduate predictive rules. The predictive rule ontology, student ontology and the part of representation implementing with backward chaining reference method. The examples of reasoning rules are shown as follows.

Backward chaining inference method:

- 1) $hasClass(Success, ?student) \leftarrow hasGAT_Score(?student, [150-300]) \wedge hasGender(?student, female)$
- 2) $hasClass(Success, ?student) \leftarrow hasGAT_Score(?student, [150-300]) \wedge hasGender(?student, male) \wedge hasGPA_2_2(?student, high)$
- 3) $hasClass(Success, ?student) \leftarrow hasGAT_Score(?student, [100-150]) \wedge hasGender(?student, female) \wedge hasGPA_2_2(?student, high)$
- 4) $hasClass(Success, ?student) \leftarrow hasGAT_Score(?student, [100-150]) \wedge hasGender(?student, male) \wedge hasGPA_2_2(?student, high)$
- 5) $hasClass(Success, ?student) \leftarrow (?student, [100-150]) \wedge hasGender(?student, male) \wedge hasGPA_2_2(?student, low) \wedge hasGPA_SC(?student, high) \wedge hasSiblings(?student, <3)$
- 6) $hasClass(Fail, ?student) \leftarrow hasGAT_Score(?student, [150-300]) \wedge hasGender(?student, male) \wedge hasGPA_2_2(?student, low)$
- 7) $hasClass(Fail, ?student) \leftarrow hasGAT_Score(?student, [100-150]) \wedge hasGender(?student, male) \wedge hasGPA_2_2(?student, low) \wedge hasGPA_SC(?student, low)$
- 8) $hasClass(Fail, ?student) \leftarrow hasGAT_Score(?student, [0-100])$
- 9) $hasClass(Fail, ?student) \leftarrow hasGAT_Score(?student, [100-150]) \wedge hasGender(?student, female) \wedge hasGPA_2_2(?student, low) \wedge hasDisease(?student, yes)$
- 10) $hasClass(Fail, ?student) \leftarrow hasGAT_Score(?student, [100-150]) \wedge hasGender(?student, male) \wedge hasGPA_2_2(?student, low) \wedge hasGPA_SC(?student, high) \wedge hasSiblings(?student, >=3)$
- 11) $Student(?student)$
- 12) $hasGAT_Score(?student, ?GAT_score)$
- 13) $hasGender(?student, ?sex)$
- 14) $hasGPA_2_2(?student, ?GPA_2_2)$
- 15) $hasGPA_SC(?student, ?GPA_SC)$
- 16) $hasDisease(?student, ?disease)$
- 17) $hasSiblings(?student, ?siblings)$

etc.

Query Example#1: $hasClass(?class, ?student)$
 $?student = "Botan"$

Reasoning Rule:

$Student(Botan)$

$hasGAT_Score(Botan, [100-150])$

hasGender(Botan, female)
hasGPA_2_2(Botan, high)
Result:
Answer: hasClass(Success, Botan)

Query Example#2: *hasClass(?class, ?student)*
?student = "Danny"

Reasoning Rule:
Student(Danny)
hasGAT_Score(Danny, [100-150])
hasGender(Danny, male)
hasGPA_2_2(Danny, high)
hasGPA_SC(Danny, high)
hasSibling(Danny, >=3)

Result:
Answer: hasClass(Fail, Danny)

SWRL as an inference Layer, is used to establish individual relationships and adaptation rule. In this paper we extend the predictive into reasoning rules. Various relationships are captured in the body of SWRL rules that described about the relation between feature and student.

The inference tree is shown in Figure 5. For example, to finding the target class result of query example#1 *hasClass(?class, ?student)* that defined above. The reasoning rule number 4, 11, 12, 13 and 14 are used in the backward chaining reasoning method. Finally, the target class "Success" will be returned. As the same inference process, In case of example#2, the target class "Fail" will be returned.

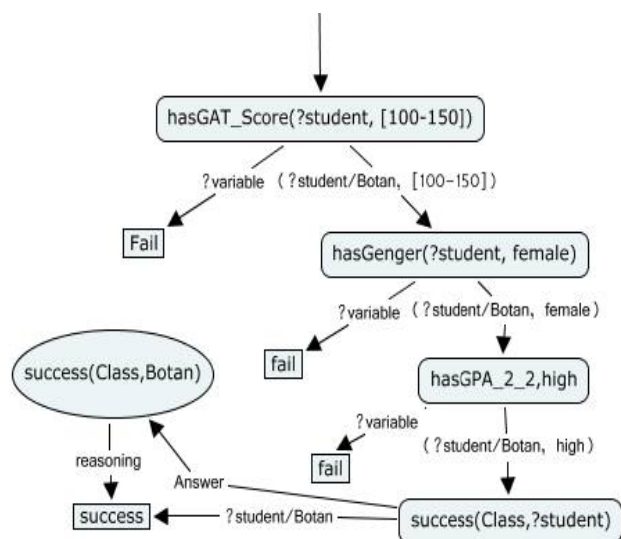


Figure 5. Example of Inference Tree

IV. CONCLUSIONS

This paper has described the system we have used to predict the results of successful graduate in bachelor degree. The classification algorithms in data mining are used to generate the classification rules. Graduated factors are identified by an experiment result. Moreover, we extend the classification rule by applying a semantic-based approach for creating a classification tree ontology. The ontology represent about the classification rules that used to enable machines to interpret and identify learner factors in process of prediction. The inference layer of classification tree ontology is based on SWRL (Semantic Web Rule Language), making a clarify separation of the program component and connected explicit modules.

In the future, we plan to make the entire implementation to train the prediction parameters itself when new training sets are imported. Also, we will use a semantic-based predictive rule to include in LMS which providing a decision support module for a current student.

REFERENCES

- [1] I. Horrocks, SWRL: A Semantic Web Rule Language-Combining OWL and RuleML, http://www.w3.org/Submission/SWRL/_2004
- [2] OWL Web Ontology Language Reference, <http://www.w3.org/TR/owl-ref>. (Accessed: June 13, 2013)
- [3] J. Han and M. Kamber, Data Mining: Concept and Techniques, Elsevier, 2006.
- [4] P. N. Tan, M. Steinbach and V. Kumar, Introduction to Data Mining, Addison-Wesley, 2006.
- [5] Protégé Ontology and Knowledge Base Framework, <http://protege.stanford.edu>. (Accessed: May 25, 2013)
- [6] M. Kantardzic, Data Mining: Introductory and Advanced Topics, Pearson Education Inc, 2003.
- [7] A. A. Freitas, Data Mining and Knowledge Discovery with Evolutionary Algorithms, Natural Computing Series, Springer: New York, 2002.
- [8] R. J. Roiger and M. W. Geatz, Data Mining A tutorial-Based Primer, Pearson Education Inc., 2003.
- [9] WEKA: <http://www.cs.waikato.ac.nz/ml/weka>.
- [10] I. Guyon, J. Weston, S. Barnhill, V. Vapnik. Gene selection for cancer classification using support vector machines. Machine Learning. 46:389-422. (2008)



Noppamas Pukkhem was born in Thailand, in 1977. She received the B.Sc. and M.Sc. degree in Computer Science from the Prince of Songkla University, Thailand, and the Ph.D. degree in Computer Engineering from Chulalongkorn University, Bangkok, Thailand, in 2010.

Since October 2004, she has been with the Department of Computer and Information Technology, Faculty of Science, Thaksin University, where she was a lecturer. She is a research assistant in College of Science and Letters, Illinois Institute of Technology (IIT), Chicago, USA from 2006 to 2007. Her current research interests include data mining, learning object technologies, AI and semantic web technologies.