Synthetic Minority Over-Sampling for Improving Imbalanced Data in Educational Web Usage Mining

Wacharawan Intayoad[1], **Chayapol Kamyod**[2], and **Punnarumol Temdee**[3]

ABSTRACT

Educational data mining is the method for extracting and discovering new knowledge from education data. As education data is often complex and imbalanced, it requires a data preprocessing step or learning algorithms in order to obtain accurate analysis and interpretation. Many studies emphasize on classification and clustering methods in order to get insight and comprehensive knowledge from education data. However, a small number of previous works exclusively focused on the preprocessing of education data, particularly on the topic of the imbalanced dataset. Therefore, this research objective is to enhance the accuracy of data classification in educational web usage data. Our study involves the application of synthetic minority over-sampling techniques (SMOTE) to preprocess the raw dataset from web usage data. The minority class is a group of the students who failed the examination and the majority class is the students who passed the examination. In our experiments, four synthetic minority over-sampling methods are applied, SMOTE, and its variants: Borderline-SMOTE1, Borderline-SMOTE2, and SVM-SMOTE, in order to balance the number of samples in the minority class. The experiments are evaluated by comparing the results from well-known classification methods that are Naive Bayesian, decision tree, and k-nearest neighbors. The study experiments with real-world datasets from education data. The results present that synthetic minority over-sampling methods are capable of improving the detection of the minority class and achieve improving classification performance on precision, recall, and F1-value.

Keywords: Education Data Mining, Imbalanced Data, Web Usage Mining

1. INTRODUCTION

Several studies in Educational data mining (EDM) are based on web-usage mining [1],[2]. It can be considered as the intelligent part of e-learning systems as it focuses on finding significant knowledge from education data [3]. Web-usage mining uses education data to discover learners' behaviors, find browsing or navigating problems, classify learning groups, and predict student performances [3],[4]. For example, the online learning systems make used of web usage mining to classify learning styles or characteristics of individual learners in order to provide suitable learning objects (LOs) and to support learners [3], [4]. These methods and techniques of web-usage mining are data-driven. It is the foundation that the success of the EDM results is strongly depending on the quality of data used, particularly the classification and prediction methods.

One of the significant problems of the real-world dataset is the imbalanced learning problem as it recurs in many domains [5–7]. Imbalanced of a dataset means that some classes significantly outnumber of examples than other classes [8]. Classification and prediction methods work poorly on imbalanced datasets. The minority class tends to be misclassified as the class boundary can be skewed towards the target class. However, often that the minority class is the important class to learn [9].

As well as in EDM, the class imbalance problem gets more and more attention, as it often faces the problem of imbalanced datasets. While students who pass or continue with their studies (normal examples) is the majority class, the examples of students who fail or leave are generally rare and from the minority class [10]. As a result, predicting or classifying a student who fails or leaves could be excessively high in incorrect prediction. However, in the education domain, we do not only focus on regular learners but also who have difficulties learning. Detecting and understanding student failure are important for educational professionals [10]. We need an effective way to detect failure students and understand their learning behavior in order to provide the appropriate assistant to them. Various methods have been proposed to solve the class imbalance problem. For example, the algorithmic approach, data preprocessing, ensemble methods and feature selection approach [9], [11]. However, there are a few attentions from previous works in EDM focused on imbalanced datasets. Moreover, detecting the characteristics of students that can influence studying failure is diffi-

Manuscript received on July 7, 2018 ; revised on December 28, 2018.

Final manuscript received on January 27, 2019.

[1,2,3] The authors are with Department of Computer Engineering, Mae Fah Lung University, Thailand., E-mail: wacharawan.int@mfu.ac.th, chayapol.kam@mfu.ac.th and Punnarumol@mfu.ac.th

cult as due to a large number of risk factors [10].

Therefore, this study proposes to solve the class imbalance problem in EDM to enhance the accuracy of classification methods for web-usage data. We particularly focus on students' performance by predicting students who pass or fail. We use the context information extracted from web-usage data as the factors for predicting student failure. The context information used in our study is the learning patterns that are the collections of the interaction between students and LOs. Our experiments are based on synthetic minority over-sampling methods (SMOTE). SMOTE generates new minority samples in order to improve the detection of a minority class and achieve improving classification performance. We deploy SMOTE, and its variants: (1) regular SMOTE, (2) Borderline-SMOTE1, (3) Borderline-SMOTE2, and (4) SVM-SMOTE to acquire an understanding of the different applicable over-sampling methods and the effect in the term of classification learning patterns. We experiment the impact of the over-sampling methods with well-known methods of classification and prediction methods in the area of personalized learning: Naive Bayesian (NB), decision tree (DCT), and k-nearest neighbors (k-NN). Furthermore, we compare the experiment results by using various metrics, including precision, recall, and F-score.

The reminders of this paper are as follows. Section 2 discusses the application of web usage mining in educational data mining as well as points out the challenges in web-usage mining when it has to cope with the raw data from online learning systems. Moreover, the section points out the advantage of SMOTE over other methods regarding the class imbalance problem. Section 3 presents the proposed methodology for coping with imbalanced datasets based on the challenges in EDM. Section 4 illustrates the results of the experiments with real-world datasets. And section 5 draws the final conclusion and discussion.

2. LITERATURE REVIEW

2.1 Educational Web Usage Mining

Web usage mining facilitates the innovation of online learning by aggregating and analyzing web usage data in a meaningful way in order to discover new knowledge [12]. In the past years, web usage mining contributed to find significant education knowledge. Such knowledge can be used to improve online learning systems, such as personalized learning in e-learning platforms, i.e., recommendation and adaptive learning services. For example, Romero et al. [1] proposed an architecture of a recommendation system. They utilized web usage mining to discover personalized recommendation links. The architecture classifies the students in one of the clusters and only uses the rules of the corresponding cluster to make the recommendations. Furthermore, Romero et al. [2] used the web usage mining for predicting final marks of students from Moodle courses. They deployed various classification algorithms. The previous studies illustrate that web-usage mining can help understanding learner behaviors and can be adapted for enhancing personalized learning in online learning platforms.

Web usage mining focuses on web usage data and opens to enhance the analysis by combining with additional information. Web usage data is the primary data source for web usage mining in which the activities of users are captured. Web usage data contains click pattern or clickstream and collects the relevant information about user interaction with the website [12]. In general, clickstream is simply stored in a chronological way that how users interact with systems. Hence, Web usage data from online learning systems contains useful information, such as the interaction between learners and LOs, visited hyperlink, number of visits, timestamp, and sequence of actions. It may be enriched with additional information, for instance, online exams, using forums, mails, discussion boards and downloaded materials [13]. Analyzing such data can help online learning platforms to provide personalized learning to learners. This type of analysis involves the automatic pattern discovery and behavior analysis, including the specification of an individual user profile [14]. Pattern discovery is often shown as the collection of the web's objects that are accessed frequently by a group of users with common interest or needs[14], and in this research, we focus on LOs. Therefore, the web usage data, which is collected on the server side, in our paper describes the student learning patterns of the usage of the LOs.

However, the raw data from online learning systems is often complex, noisy, and imbalance. Some of the encountered problems are:

- Imbalanced data: Datasets are considered to be imbalanced when the classification categories are not approximately equally represented [12]. This scenario leads to bias prediction and misleading accuracy. Many studies point out that several standard classifiers work poorly mainly due to the skewed distribution, given by the imbalance ratio. Imbalance ratio is defined as the ratio of the number of majority class instances to the number of minority instances[9, 15, 16]. As well as in online learning systems, the scenario of imbalanced data could happen in education datasets [11]. For example, a good e-learning system can provide effective learning assistance. As a result, the different number of students who pass and students who fail the examination is significantly high. Data instances which represent minority class is insufficient. So if we try to classify the students who pass the exam (majority class), the accuracy of prediction will be very high. But that is not the objective of classification in e-learning which also aim to observe the minority class as well.
- Noise: Noise, in this case, can be determined as

the unintended of user clicks. In the case of the randomness of user clicks means that learners just click on any learning objects randomly for searching specific content or randomly click without any purpose of studying. In order to analyze the intended learning behaviors, it has to exclude these random clicks. Moreover, in the situation of imbalanced data, when the imbalance ratio is very high and the number of minority class is limited, the presence of noise has a greater impact on the minority class [17].

• Flexible environment: Online learning systems provide flexible learning environments which allow learners to study in any preferences of sequences. Moreover, the flexible environment creates an exponential number of possible learning paths which makes complexity to identify and analyze learning behaviors.

In order to manage the aforementioned challenges from education data, web usage data requires heuristics algorithms and methods for data preprocessing which are subjected to the characteristics of raw data and the objective of the analysis [4].

2.2 Imbalanced Datasets

Data preprocessing step is the important task in data mining as well as in web usage mining. As the quality of input data determines the accuracy of the data interpretation and analysis [2]. Data preprocessing is able to reduce the complexity of web usage mining and its tasks are varied and related to the goals of mining [18]. The data preprocessing step in data mining may involve many tasks, such as data gathering, data integration, data cleaning, data filtering, and data transformation. Yet a number of unique challenges of web usage mining have led to a variety of algorithm and heuristics techniques, such as data fusion, user and session identification, page view identification, data reduction, path completion, and session identification [14], [19–21]. However, it is not intended for this paper to describe all available methods. This section discusses the data preprocessing and algorithms associated with the class imbalance problem.

Imbalanced dataset means that one of the classes has a significant number of examples than the other [22][23]. Imbalanced datasets may be associated with binary classes or multiple classes. We call the class which outnumbers the examples as majority class, while we call the class with has less number of examples as minority class [8]. Most of the standard algorithms of classification often misclassify the minority class regarding imbalanced dataset. In the work of [9] summarized the main reasons behind this imbalanced classification problem. The first reason is related to the use of inappropriate global performance measures. For example, standard accuracy rate may provide an advantage for the majority class. The second reason is related to the classification rules that predict the positive class are often highly specific and they are discarded general rules, i.e. those are associated with a negative class. The last reason is related to the small size of minority class examples. These minority examples may be identified as noise. As a result, they could be incorrectly rejected by the classifier. On the contrary, few real noisy examples can affect the accuracy of predicting minority class since it has fewer examples.

Moreover, there are some characteristics of imbalanced datasets that may impact the quality of classifiers, such as the problem of small disjuncts, and class overlapping. The small disjuncts problem arises minority class predictions tending to be formed from fewer training examples than majority class predictions [9], [24] and minority class is underrepresented with the respect of the majority samples. The problem of class overlapping occurs when two classes have a similar number of training data points in a region of data space [9]. The problem leads to the same a probabilities in the cover lapping area and makes it difficult to distinguish between two classes.

Many methods and algorithms have been proposed to cope with imbalanced classification problem. In the work of [25] categorized methods for dealing with class imbalanced into three types for their experiments (1) resampling, (2) down-sizing, and (3) learning by recognition. Resampling methods focus on resampling the minority class until the number of its examples has as many as majority class. Down-sizing methods focus on eliminating the examples of majority class until it matches the size of the other class. Moreover the last one is learning by recognition methods. These methods based on the classification approach. The idea is to train the network to learn to recognize one of the two classes. Once it learned, it is can make a decision to accept or reject an example whether it belongs to the class on which it was trained or not. In the work of [9] also categorized the methods for coping with the class imbalance problem. They categorized methods into three categories. The first one is the preprocessing. This category focuses on methods involving in the data preprocessing phase, such as data sampling methods (both oversampling and down-sizing). The second one is cost-sensitive learning. It takes into account the variables cost of misclassification [26], [27] and introduces it to learning algorithms. Also the last one is ensemble methods. These methods try to improve upon the single class classifier by combining and inducing several classifiers to get a new classifier that outperforms every one of them.

The work of [25], demonstrated the experiments comparing the performance of the several approaches related to class imbalance problem, such as resampling methods, downsizing methods, and learn by recognition methods. The several datasets were created with the various combination of three different

dimensions (concept complexity, training set size, and degree of imbalance). The experiments illustrated that the resampling and downsizing methods were effective, particularly as the concept of complexity getting larger. On the contrary, other methods, such as learning by recognition methods, do not have an advantage. As well as in the work of [9], they analyzed the behavior of imbalanced learning methods. They discovered that the preprocessing as over-sampling methods, such as SMOTE and its variants appear to be the robust algorithms and obtain very good results. Therefore SMOTE and its variants can be considered as a standard approach for imbalanced datasets.

Several works suggest that over-sampling or resampling methods perform very effective in the imbalanced class problem. Nonetheless, the random resampling method has the major drawbacks regarding increasing the likelihood of occurring overfitting as it duplicates of existing examples [9]. However, SMOTE is a more sophisticated method which can deal with the mentioned problem [9]. It takes each minority class samples and generates synthetic examples along the line segments joining of the k minority class nearest neighbours. The k nearest neighbours are randomly chosen. In this way, the additional of information within the minority class examples from new synthetics examples allows more information from larger cluster to help classifiers to separate both classes. However, such over-sampling techniques, including SMOTE algorithms, may creates the problem of over generalization. For example, SMOTE generates the same number of samples for each original minority examples without considering the neighbouring examples leading to the occurrence of overlapping between classes. AS a result, various variants of SMOTE have been proposed in order to cope with this problem.

This study based on SMOTE algorithm and its variants in order to cope with class imbalance problem in educational web usage data. As it has been proved from many studies that it is one of the effective methods for dealing with the class imbalance problem. Moreover, SMOTE and its variants are very efficient in the aspect of computing time when comparing with other ensemble methods.

3. METHODOLOGY

Our research objective is based on improving the accuracy of classifying and predicting learning performance in online learning systems. The study is developed under the e-learning system of our university. We aim to classify students' learning patterns in order to predict the result of studying (pass/fail). The learning patterns can be used as an inference for student learning behaviors, as learning behaviors can be used as the factor of student success and failure.

The learning patterns in the study are gathered from web usage data which is recorded in web server log files from the e-learning system. The server logs record the interaction of students with the LOs. The student profiles and examination results from the learning management system (LMS) are integrated for analyzing learning patterns. Our proposed methodology is based on supervised learning because the class labels are provided in the data. The methodology is composed of three phases: (1) data preprocessing, (2) classification process, and (3) classification quality.

3.1 Data Preprocessing

As mentioned in section 2, our work focuses on improving accuracy when education data is imbalanced between positive class and negative class. And we also highlight the problems of the noise and outliers. We proposed four approaches to these two challenges.

3.1.1 Imbalanced data with the SMOTE

Our methods to deal with imbalanced data to improve imbalance binary classification are based on over-sampling method: Synthetic Minority Oversampling Technique (SMOTE) [28]. Since they are robust among many different situations [29], [30].

Let P be the minority class and N be the majority class, the amount of new samples of P is S_P, k be a number of nearest neighbours, and algorithm of SMOTE is as following:

1. Calculate k nearest neighbors for examples in P
2. Randomly select example p_i in P, and its nearest neighbors
3. Randomly select nearest neighbors nn of p_i
4. Calculate difference dif for every attributes between the p_i and nn.
5. Create new synthetic minority example by $dif \times gap$, where gap is a random number between 1 and 0
6. Repeat step 2 to 5 until the number of new generated examples equal to S_P.

3.1.2 Borderline-SMOTE1 and Borderline-SMOTE2

These two methods are based on SMOTE which focuses on minority over-sampling. However, they are different from SMOTE by not randomly generating from entire minority examples. Instead, they generate samples near the borderline. The borderline is generated from minority examples that are considered as $DANGER$ [31].

Let minority class is P and majority class is N, $p_1, p_2, \ldots, p_{num}$ be the members of P,$n_1, n_2, \ldots, n_{num}$ be the members of N, p_{num} and n_{num} are the number of class member in P and N respectively. For every example $p_i (i = 1, 2, \ldots, p_{num})$ in P is calculate its m nearest neighbors. The number of N examples

among the m nearest neighbors is donated by m'. p_i is considered to be DANGER p'_i, if $m/2m' < m$.

$$DANGER = \{p'_1, p'_2, \ldots, p'_{num}\} \tag{1}$$

Borderline-SMOTE1 creates synthetic positive examples from $DANGER$. This step is similar to SMOTE.

For Borderline-SMOTE2, It is not only generating synthetic positive examples from borderline but also from its nearest negative neighbours in N. The new example is calculated by the difference between it and its nearest neighbours multiplied with the random number between 0 and 0.5. By this way, the new generated examples are closer to P.

3.1.3 SVM-SMOTE

Support Vector Machine (SVM) is the classification method. It has been successfully used in many applications. The objective of SVM is to find the maximum margin or optimal hyperplane to separate the positive and negative classes [23]. It is based on the linear learning system. Let $\{(x_i, y_i)\}$ be the set of training data, where x_i is a training instance, $i = 1, \ldots, N$. Each training instance is called an input vector. y_i is a class label, where $i \in \{1, -1\}$, 1 denotes to the positive class and -1 denotes to the negative class. The bold face letter denotes for all vectors. w is the weight vector, b is bias. The linear function of SVM is as follow.

$$F(\mathbf{x}) =< \mathbf{w} \times \mathbf{x} > +b \tag{2}$$

If the input vector x_i has $F(x) \geq 0$ then it is assigned to the positive class and it will be assigned to the negative class, if $F(x) < 0$.

$$\begin{aligned} y_i = 1, if < \mathbf{w} \times \mathbf{x} > +b \geq 0 \\ y_i = -1, if < \mathbf{w} \times \mathbf{x} > +b < 0 \end{aligned} \tag{3}$$

The hyperplane of SVM is

$$< \mathbf{w} \times \mathbf{x} > +b = 0 \tag{4}$$

It is the separation line between positive and negative training instances. We can rescale hyperplane to $< \alpha\mathbf{w} \times \mathbf{x} > +\alpha b = 0$ for α is a real positive number. The maximal margin hyperplane is the largest margin between hyperplanes which is called decision boundary.

In order to deal with imbalanced datasets, instead of over-sampling on the entire minority class, SVM-SMOTE approach focuses on the decision boundary. New minority instances are randomly created along the line of joining each minority class support vector. Interested readers could find the detail in [32].

3.2 Classification

There are a large number of classification methods; the paper focuses on the basic task that is a stepping stone to more advanced construction. We select three classifications to predict students' performance from their learning behaviors: (1) Naive Bayesian, (2) decision tree, and (3) k-nearest neighbor.

Naive Bayesian Classification (NB) is a supervised learning that is based on a probabilistic point of view and Bayes' theorem. It is an independent assumption between predictors. It is very beneficial for a large dataset, easy to understand and well performing mostly. The equation (5) is the NB.

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(x)} \tag{5}$$

Let D be a training set of tuples and associated to the class label. Each tuple is an n-dimension attribute, $X = (x_1, x_2, \ldots, x_n)$. C is the set of class and m is the number of class,$C_1, C_2, \ldots, C_m$. Let $P(C_i|X)$ is the posterior probability of target class given predictor. Classifier will predict X belong to the class having the maximum posterior probability [33].

Decision trees (DCT) are predictive models based on the probability distribution of occurrence. They are composed of nodes and arcs. There are two types of nodes: (1) leaf node, and (2) root node. The leaf node is a decision node that presents the possible attributes. The root node is the outcome node that owns the attributes [14]. Generally, the learning algorithms are done by the divide-and-conquer strategy that iteratively partitions the training datasets to produce the tree. Each iterative chooses the best attribute to the partition data at the current node. The best attribute is based on the impurity function, such as C4.5 which is the widely used decision tree learning algorithm in various domains [34].

K nearest neighbors (k-NN) is a lazy learning method which means that the learning will occur when it needs to classify the new sample. The algorithm compares the new instance x_i with every samples in the training dataset D by computing the similarity between them. The similarity is derived from distance/similarity functions which are chosen based on applications and the nature of data [14]. The k is number of the most nearest neighbors of x_i in D. The most frequent class among k nearest neighbors of x_i is selected.

3.3 Classification Quality

Classifiers are required to be evaluated for accuracy. It is crucial to know the approximate accuracy of a classifier in order to ensure that it can be used in the real-world tasks. Several metrics have been used to evaluate the quality of classification. Performance of classification and machine learning are often eval-

uated by a confusion matrix and it can be used to investigate the minority class. Confusion matrix that is presented in Table 1 where TP is a true positive, FN is the false negative, FP is the false positive, and TN is the true negative.

Table 1: *Confusion Matrix.*

	Classified positive	Classified negative
Actual Positive	TP	FN
Actual Negative	FP	TN

This study performs three measurements: precision, recall ,and F-score. The first two are based on the primary measure of classifiers, classification accuracy. The classification accuracy is presented in equation 1 [14]. And some studies used the error rate which is 1-accuracy.

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{6}$$

Based on a confusion matrix, we can derive precision p and recall r which are suitable in such imbalance data application. p and r of the positive class are defined as follows:

$$p = \frac{TP}{TP + FP} \tag{7}$$

$$r = \frac{TP}{TP + FN} \tag{8}$$

Where p is the number of correctly classified of positive instances divided by the total number of instances that are classified as positive. And r is the number of correct classified of positive instances divided by the total number of actual positive instances in the test datasets. Furthermore, the F-score is used to compare different classifiers as in practice the high precision often happens at the expense of recall and vice versa.

$$F - score = \frac{2 \times precision \times recall}{precision + recall} \tag{9}$$

4. EXPERIMENTS AND RESULTS

We have carried out preprocessing data with imbalanced datasets in order to evaluate the performance of them with the different classification algorithms as mentioned in section 3.2. The data sources that are used in this study is from the general education course in Mae Fah Luang University. The course is mandatory for every student from different background knowledge and interest. The course is composed of seven modules, and each module is composed of several LOs. When the students finish studying for each module, there will be an examination to assess the learning objective. The information of the examination results is recorded in the database. And in order to pass the exam students require 50% of the full score. In addition, student data contains many types of information, such as student ID, and major. Fig. 1 depicts the class diagram of data in the e-learning database and the Learning Management system (LMS).

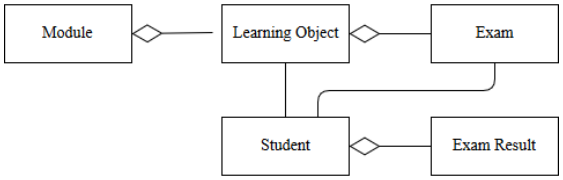

Fig.1: *Class diagram presenting the relationship of objects in the e-learning system.*

This study is based on the information gathered from web usage data and LMS database; we focus on two online activities: (1) Learning patterns from clickstream and (2) examination result. As this study focuses on web usage mining, Table 2 presents the user interaction data from the web usage data. Each row represents each event that the student interacted with the LO. For example, log ID 1 shows a student ID 001 with 'IP 1.1.20.235 'accessing the resource: '/learning.php?fid=257 'at '2015-02-15 08:02:44 'on the e-learning systems. The last part of the resource path identifies the LO, in this case, is LO257. Moreover, each path has a one-to-one correspondence with a pageview which means that an HTML file has single framed sites as well as LO. Our log data identify student ID; as a result in our case, we do not need to process the user identification task. However, the log data does not contain the information from the client machine for instance browser type, and previous navigated location.

Table 2: *Example of weblog.*

Log ID	Student ID	path	IP	Timestamp
1	001	/lo.php?fid=257	1.1.20.235	2015-02-15 08:02:44
2	002	/lo.php?fid=264	1.1.30.10	2015-02-15 09:12:25
3	001	/lo.php?fid=253	1.1.20.235	2015-02-15 08:15:29
4	003	/lo.php?fid=265	1.1.30.15	2015-02-15 10:23:16
5	001	/lo.php?fid=255	1.1.20.235	2015-02-15 08:17:08
6	001	/lo.php?fid=257	10.1.27.179	2015-02-16 17:32:25

Later the data from server log is integrated with the LMS database in order to identify the result of the examination. Table 3 demonstrates the integrated data. The first column presents student ID; the second one presents the examination result: pass and fail, the rest of the columns present a number of times that students access to the LOs. For instance, the first row presents that student ID 001 who passed the examination accessed to LO255 two time, and LO257 one time and did not access to LO256.

Table 3: *Example of Integrated data.*

Student ID	Examination Result	LO255	LO256	LO257	LO...
001	pass	2	0	1	...
002	pass	1	1	1	...
003	fail	2	0	1	...

We deploy two datasets from real-world education data: dataset A and dataset B. The dataset A is composed of 1,316 students who passed the exam and 37 students who failed the exam. There are 79 LOs in dataset A. Dataset B is composed of 390 students who passed the exam and 29 students who failed the exam. And there are 21 LOs in dataset B.

4.1 Preprocessing Imbalanced Data

In the case of our study, we focus on two classes as students who passed and who failed the examination. Generally, these two classes are called the positive class and negative class. In our study, the positive class represents the minority class that is failed students P, and the negative class represents the majority class that is passed student N.

Table 4 presents the number of examples of the original datasets and after applied synthetic minority over-sampling techniques. The first row is the original training examples. It shows that the dataset A has a high degree of imbalance ratio between majority and minority as approximately 36:1. For regular SMOTE, Borderline-SMOTE1 and Borderline-SMOTE2, the number of minority samples are generated to the same number of N examples which are 1,316. On the contrary, SVM-SMOTE generated less new minority samples, 699 samples, as it focuses on creating new samples within the decision boundary.

Table 4: *Presents the number of data instance in class P and N from the original training dataset A and oversampled datasets.*

Dataset	class P(failed student)	class N (passed student)
Original	37	1,316
SMOTE	1,316	1,316
Borderline-SMOTE1	1,316	1,316
Borderline-SMOTE2	1,316	1,316
SVM-SMOTE	699	1,316

Table 5 presents the number of examples of the original dataset B and after applied synthetic minority over-sampling techniques. The first row is the original training examples. It shows that the dataset B is also has a high degree of imbalance, the imbalance ratio of the majority to the minority is approximately 13:1. For regular SMOTE, Borderline-SMOTE1 and 2, the number of minority samples are generated to the same number of N examples which are 390. On the contrary, SVM-SMOTE generated less new minority samples, 219 samples, as it focuses on creating new samples within the decision boundary. Comparing dataset A and B in the regards of sample size, dataset A has bigger samples size in number. However, dataset set have higher imbalance ratio.

Table 5: *Presents the number of data instance in class P and N from the original training dataset B and oversampled datasets.*

Dataset	class P (failed student)	class N (passed student)
Original	29	390
SMOTE	390	390
Borderline-SMOTE1	390	390
Borderline-SMOTE2	390	390
SVM-SMOTE	216	390

To understand the dataset, principal component analysis (PCA) is used for virtualization [35]. With the PCA, we converted the dataset to 2-dimensional space and normalized the value of the features. Be noted that during the process of converting, the datasets lose some variances. Fig. 2, 3, 4, 5, and 6 depict the samples of the original training dataset A, SMOTE, Borderline-SMOTE1, Borderline-SMOTE2, and SVM-SMOTE, respectively. The x-axis and y-axis present principle component 1 and 2 respectively. And the yellow dots present minority class and the blue dots present majority class.

Fig. 2 depicts the original dataset A, most of the data instances are in the same area at x-axis between -2 and 0, and the y-axis between -2 and 2. It has an area of overlapping between two classes only in the most left region and it is likely to have a low degree of overlapping.

Fig. 3 presents PCA from SMOTE. We can observe that the newly generated samples are randomly created all possible between minority instances.

Comparing with the SMOTE (Fig.3), Borderline-SMOTE1(Fig. 4) generates the new minority samples between the samples in danger and the nearest neighbors of majority samples.

For the Borderline-SMOTE2 (Fig. 5), it also uses the samples in DANGER to generate new samples. However, instead of generating new samples between the nearest neighbors of majority samples, it randomly selects the nearest neighbors from all classes. As we can observe, the new minority samples are in the broader area in the left region.

The last one is SVM (Fig. 6), the new minority samples are generated along the line of joining each minority class support vector. Therefore, the number of new samples is less than other methods.

SMOTE, Borderline-SMOTE1, and SVM-SMOTE have slightly the same patterns of newly created samples. The new samples are created in about the same area as the original data. For the Borderlines-SMOTE 2, there are generate synthetic positive examples from its nearest negative neighbor in major-

ity class. Thus, the new samples are generated in the broader range of dimensional space.

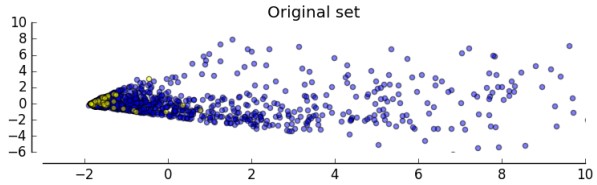

Fig.2: *Dot chart presenting original data instances from dataset A.*

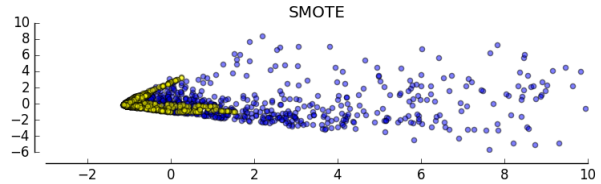

Fig.3: *Dot chart presenting data instances from SMOTE from dataset A.*

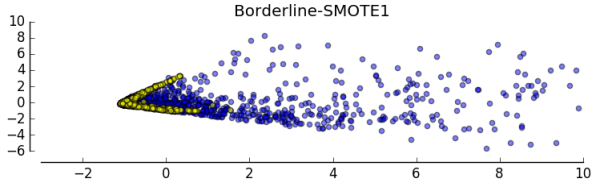

Fig.4: *Dot chart presenting Borderline-SMOTE1 instances from dataset A.*

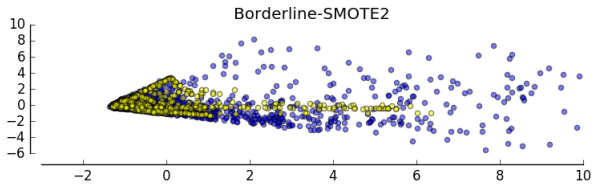

Fig.5: *Dot chart presenting Borderline-SMOTE2 instances from dataset A.*

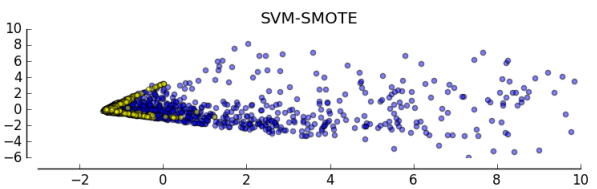

Fig.6: *Dot chart presenting SVM-SMOTE instances from dataset A.*

Fig. 7, 8, 9, 10, and 11 depict the samples of the original training dataset B, SMOTE, Borderline-SMOTE1, Borderline-SMOTE2, and SVM-SMOTE, respectively.

Fig. 7 depicts the original dataset, most of the data instances are in the same area at x-axis between -2 and 2, and the y-axis between -4 and 4. This situation presents that dataset B has small disjuncts for minority class. The positive examples are underrepresented with respect to the positive examples in all area of the region.

Fig. 8 is a plot from SMOTE. We can observe that the newly generated samples are randomly created all possible between minority instances. As well as in Fig. 9, Borderline-SMOTE1 generates the new minority samples between the samples in danger and the nearest neighbors of majority samples. Both SMOTE and Borderline-SMOTE1 have a high degree of class overlapping, particularly in the left region.

Borderline-SMOTE2 (Fig. 10), we observe that new minority samples create likely in the same area of SMOTE and Borderline-SMOTE1, As the cause of small disjuncts and a high degree of class overlapping of the dataset B. It uses the samples in DANGER to generate new samples between the nearest neighbors of majority samples.

The last one is SVM (Fig. 11). As the new minority samples are generated along the line of joining each minority class support vector. Therefore, the number of new samples is less than other methods.

For dataset B, SMOTE, Borderline-SMOTE1, Borderline-SMOTE2, and SVM-SMOTE have slightly the same patterns of newly created samples. As the larger number of new minority samples created, the more degree of class overlapping.

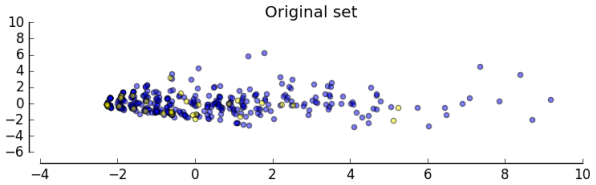

Fig.7: *Dot chart presenting original data instances from dataset B.*

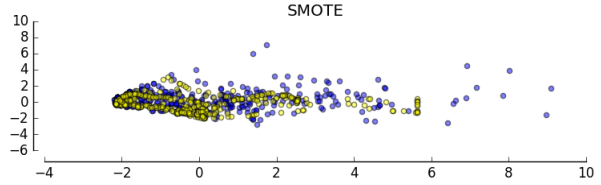

Fig.8: *Dot chart presenting data instances from SMOTE from dataset B.*

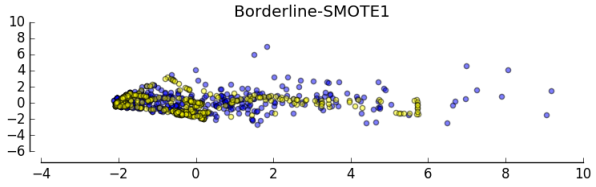

Fig.9: *Dot chart presenting data instances from Borderline-SMOTE1 from dataset B.*

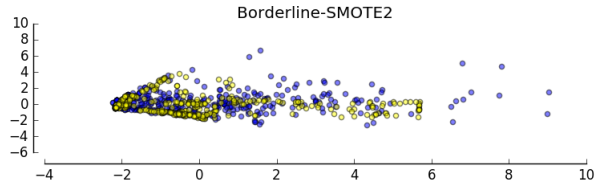

Fig.10: *Dot chart presenting data instances from Borderline-SMOTE2 from dataset B.*

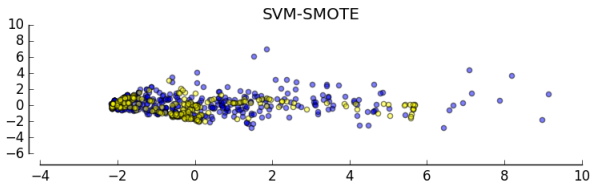

Fig.11: *Dot chart presenting data instances from SVM-SMOTE from dataset B.*

4.2 Classification Quality

We compare the results of synthetic minority over-sampling methods through precise and recall of all classes as well as the F-value. Precise and recall rate reflects the performance of over-sampling methods through the classification performance. Three widely used machine learning algorithms; NB, DCT, and k-NN, are used in our experiments as we overviewed in section 3.2.

Table 6, 7, and 8 demonstrate the experimental results from dataset A, while table 9, 10, and 11 demonstrate the experimental results from dataset B. The first column presents the over-sampling methods and the second to fourth column present precision, recall, and F1-score for minority class P, respectively. The fifth to seventh column present precision, recall and F1-score for majority class N, respectively.

Table 6 presents the experimental results of NB classification performance. It is obvious that all the over-sampling methods significantly improve the precision and recall rates of minority class P. When comparing with the original dataset, the best performance of precision and recall rate for NB is Borderline-SMOTE1. For the precision, recall and F1-score of the majority class N, the rates are slightly decrease for all over-sampling methods.

Table 6: *Classification quality results from NB, dataset A.*

	P Pre-ci-sion	P re-call	P F1-score	N Pre-ci-sion	N Re-call	N F1-score
Original	0.08	0.19	0.12	0.98	0.94	0.96
SMOTE	0.70	0.89	0.78	0.85	0.62	0.72
Borderline1	**0.74**	**0.95**	**0.84**	**0.93**	**0.67**	**0.78**
Borderline2	0.70	0.89	0.78	0.85	0.61	0.71
SVM	0.64	0.88	0.74	0.91	0.71	0.80

Table 7 presents the classification quality from DCT. Comparing with the original dataset, the precision, recall, and F1-score are increase. All the four over-sampling methods have the equivalent performance regarding on the classification quality. The four over-sampling methods do not have any significant effect on the classification quality over the majority class N.

Table 7: *Classification quality results from DCT, dataset A.*

	P Pre-ci-sion	P re-call	P F1-score	N Pre-ci-sion	N Re-call	N F1-score
Original	0.91	0.78	0.84	0.99	1.00	1.00
SMOTE	1.00	0.99	1.00	0.99	1.00	1.00
Borderline1	1.00	0.99	1.00	0.99	1.00	1.00
Borderline2	1.00	0.99	1.00	0.99	1.00	1.00
SVM	1.00	0.99	1.00	0.99	1.00	1.00

Table 8 presents the classification quality from k-NN. Unlike the DCT, all of the over-sampling methods behaves excellent for improving the classification quality of minority class P. Compare with the original datasets, the best improvement of precision and F1-score of minority class P is Borderline1. For recall rate of minority class P, SMOTE and Borderline-SMOTE2 have the best performance. The precision, recall and F1-score are barely affect the quality by the over-sampling methods.

Table 8: *Classification quality results from k-NN, dataset A*

	P Pre-ci-sion	P re-call	P F1-score	N Pre-ci-sion	N Re-call	N F1-score
Original	0.00	0.00	0.00	0.97	1.00	0.99
SMOTE	0.87	**0.97**	0.92	**0.97**	0.86	0.91
Borderline1	**0.9**	0.96	**0.93**	0.96	0.89	0.92
Borderline2	0.88	**0.97**	0.92	**0.97**	0.86	0.91
SVM	0.84	0.95	0.89	**0.97**	**0.9**	**0.93**

Table 9 presents the experimental results of NB classification performance with dataset B. It is obvious that all the over-sampling methods improve the precision and recall rates of the minority class P. However, the difficulties for imbalanced classification emerges. The small disjuncts of the minority class P

effect the behavior of learning algorithm. Moreover, as the dataset B has a high degree of overlapping that lead to the result of a correct discrimination between both classes. NB classification accuracy on the majority class N decrease when comparing with the original dataset.

Table 9: *Classification quality results from NB, dataset B*

	P Precision	P recall	P F1-score	N Precision	N Recall	N F1-score
Original	0.00	0.00	0.00	0.93	1.00	0.96
SMOTE	0.57	0.56	0.57	0.57	0.58	0.58
Borderline1	0.59	0.57	0.58	0.59	0.61	0.60
Borderline2	0.55	0.50	0.53	0.54	0.59	0.56
SVM	0.59	0.26	0.36	0.69	0.90	0.78

Table 10 presents the classification quality from DCT. Comparing with the original dataset, the precision, recall, and F1-score are increase. When applying DCT to the original data, the classifier could not recognize the positive class. Applying synthetic minority over-sampling methods boosts the performance of classification of DCT, particularly on the minority class P. All the four over-sampling methods have the equivalent performance regarding the classification qualities. Similarly to dataset A, the four over-sampling methods do not have any significant effect on the classification quality over the majority class N.

Table 10: *Classification quality results from DCT, dataset B*

	P Precision	P recall	P F1-score	N Precision	N Recall	N F1-score
Original	0.00	0.00	0.00	0.98	**0.99**	**0.99**
SMOTE	**0.99**	0.97	**0.98**	0.97	**0.99**	0.98
Borderline1	**0.99**	0.96	**0.98**	0.96	**0.99**	0.98
Borderline2	0.97	**0.99**	**0.98**	**0.99**	0.97	0.98
SVM	0.98	0.96	0.97	0.98	**0.99**	0.98

Table 11 presents the classification quality from k-NN. All of the over-sampling methods behave excellently for improving the classification quality of minority class P. Comparing with the original datasets, the best improvement of precision and F1-score of minority class P is Borderline1. For recall rate of minority class P, SMOTE has the best performance. For the majority class N, the precision is slightly improve. For recall and F1-score are slightly decrease.

The experiments have been conducted with two real-world datasets from educational web usage data. The results from the classifiers tested with original data show that the classification accuracy of the majority class (passed students) is very high, while the accuracy of minority class (failed students) is weak. The results should be recognized that erroneous recognition of failed students does not give any benefit to improve learning process in online learning systems. As we fail to predict who has the potential to fail, we are not able to provide the appropriate support to the students.

Table 11: *Classification quality results from k-NN, dataset B*

	P Precision	P recall	P F1-score	N Precision	N Recall	N F1-score
Original	0.00	0.00	0.00	0.93	**1.00**	**0.96**
SMOTE	0.8	**0.94**	0.87	0.93	0.77	0.84
Borderline1	**0.83**	0.92	**0.88**	0.91	0.82	0.86
Borderline2	0.81	0.93	0.86	0.92	0.78	0.84
SVM	0.77	0.89	0.82	**0.94**	0.85	0.89

After applying SMOTE and its variants with dataset A, it illustrates that these methods are capable of improving the classification quality for minority class P, particularly for k-NN and NB. For DCT, synthetics minority over-sampling methods does not perform any advantage. For dataset B which has less imbalance ratio comparing wit dataset A, but smaller in a number of examples, disjuncts, and has a high degree of class overlapping. The SMOTE and its variants significantly improve the quality of classification for minority class P. However, for NB, the characteristics of dataset B cause erroneous prediction in NB.

5. CONCLUSION AND DISCUSSION

One particular problem in education data is the imbalanced class as the distribution of the examples in the datasets is not equal for all classes. For example, the number of students who passed the examination is often more than a number of students who failed the examination. This typical distribution of examples in datasets leads to misleading interpretation. In this correspondence, this study emphasized on acquiring a deeper understanding of the different applicable of over-sampling methods regarding the class imbalance problem and the effect in the term of classification learning patterns. Moreover, we have proposed learning pattern as the factor that could impact the failure of study. We compare the synthetic minority over-sampling method and its variance approaches by using various metrics, including precision, recall, and F-score. In addition, our study identifies that the imbalanced data problem in education data influences the results of classification, particularly on the minority class.

One of the remarkable points in this study is that the synthetic minority over-sampling methods, SMOTE, Borderline-SMOTE1, Borderline-SMOTE2, and SVM-SMOTE, are able to achieve on improving the precision, recall, and F1-score for minority class particularly for NB and k-NN classification. This demonstrates that synthetic over-sampling builds larger decision regions that contain nearby mi-

nority class examples can improve the accuracy of classification. However, the imbalance ratio does not have significant effect on the performance of classifiers after applying synthetic minority over-sampling methods. On the contrary, a degree of class overlapping, and small disjuncts have more impact to the performance of classifiers, particularly on NB. Furthermore, we have shown that learning patterns form web usage data, the interaction between students and LOs, can be very useful for predicting students' performance.

In the future experiments can be extended to different characteristics of datasets and obtaining additional data. Moreover, developing over existing methods or integrated them should be considered in order to improve the performance on the existing methods.

References

[1] C. Romero, S. Ventura, A. Zafra, and P. de Bra, "Applying Web usage mining for personalizing hyperlinks in Web-based adaptive educational systems," *Comput. Educ.*, vol. 53, no. 3, pp. 828-840, Nov. 2009.

[2] C. Romero, P. G. Espejo, A. Zafra, J. R. Romero, and S. Ventura, "Web usage mining for predicting final marks of students that use Moodle courses," *Comput. Appl. Eng. Educ.*, vol. 21, no. 1, pp. 135-146, 2013.

[3] H. Ba-Omar, I. Petrounias, and F. Anwar, "A framework for using web usage mining to personalise e-learning," in *Advanced Learning Technologies, 2007. ICALT 2007. Seventh IEEE International Conference on*, 2007, pp. 937-938.

[4] M. Munk and M. Drlk, "Impact of different pre-processing tasks on effective identification of users' behavioral patterns in web-based educational system," *Procedia Comput. Sci.*, vol. 4, pp. 1640-1649, 2011.

[5] C. Tsai, L. Chang, and H. Chiang, "Forecasting of ozone episode days by cost-sensitive neural network methods," *Sci. Total Environ.*, vol. 407, no. 6, pp. 2124-2135, 2009.

[6] C. Schumacher and D. Ifenthaler, "Features students really expect from learning analytics," *Computers in Human Behavior*, vol. 78, pp.397-407, 2018.

[7] R. Batuwita and V. Palade, "microPred: effective classification of pre-miRNAs for human miRNA gene prediction," *Bioinformatics*, vol. 25, no. 8, pp. 989-995, 2009.

[8] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Special issue on learning from imbalanced data sets," *ACM Sigkdd Explor. Newsl.*, vol. 6, no. 1, pp. 1-6, 2004.

[9] V. Lopez, A. Fernandez, S. Garcia, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," *Inf. Sci.*, vol. 250, pp. 113-141, 2013.

[10] C. Marquez-Vera, A. Cano, C. Romero, and S. Ventura, "Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data," *Appl. Intell.*, vol. 38, no. 3, pp. 315-330, 2013.

[11] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "A comparative study of data sampling and cost sensitive learning," in *Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on*, pp. 46-52, 2008.

[12] R. Cooley, B. Mobasher, and J. Srivastava, "Web mining: Information and pattern discovery on the world wide web," in *Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on*, pp. 558-567, 1997.

[13] H. M. Truong, "Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities," *Comput. Hum. Behav.*, vol. 55, Part B, pp. 1185-1193, Feb. 2016.

[14] B. Liu, "Web data mining: exploring hyperlinks, contents, and usage data," *Springer Science and Business Media*, 2007.

[15] V. Garcia, J. S. Sanchez, and R. A. Mollineda, "On the effectiveness of preprocessing methods when dealing with different levels of class imbalance," *Knowl.-Based Syst.*, vol. 25, no. 1, pp. 13-21, 2012.

[16] N. Garcia-Pedrajas, J. Perez-Rodrguez, M. Garcia-Pedrajas, D. Ortiz-Boyer, and C. Fyfe, "Class imbalance methods for translation initiation site recognition in DNA sequences," *Knowl.-Based Syst.*, vol. 25, no. 1, pp. 22-34, 2012.

[17] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM Sigkdd Explor. Newsl.*, vol. 6, no. 1, pp. 7-19, 2004.

[18] C. G. Marquardt, K. Becker, and D. D. Ruiz, "A pre-processing tool for web usage mining in the distance education domain," in *Database Engineering and Applications Symposium, 2004. IDEAS'04. Proceedings. International*, pp. 78-87, 2004.

[19] R. Cooley, B. Mobasher, and J. Srivastava, "Data preparation for mining world wide web browsing patterns," *Knowl. Inf. Syst.*, vol. 1, no. 1, pp. 5-32, 1999.

[20] N. K. Tyagi, A. K. Solanki, and S. Tyagi, "An algorithmic approach to data preprocessing in web usage mining," *Int. J. Inf. Technol. Knowl. Manag.*, vol. 2, no. 2, pp. 279-283, 2010.

[21] G. T. Raju and P. S. Satyanarayana, "Knowledge discovery from web usage data: Complete preprocessing methodology," *Int. J. Comput. Sci. Netw. Secur.*, vol. 8, no. 1, pp. 179-186, 2008.

[22] S. Ertekin, J. Huang, L. Bottou, and L. Giles,

"Learning on the border: active learning in imbalanced data classification," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pp. 127-136, 2007.

[23] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429-449, 2002.

[24] G. M. Weiss and F. Provost, "Learning when training data are costly: The effect of class distribution on tree induction," *J. Artif. Intell. Res.*, vol. 19, pp. 315-354, 2003.

[25] N. Japkowicz, "The class imbalance problem: Significance and strategies," in *Proc. of the Int'l Conf. on Artificial Intelligence*, 2000.

[26] P. Domingos, "Metacost: A general method for making classifiers cost-sensitive," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 155-164, 1999.

[27] B. Zadrozny, J. Langford, and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pp. 435-442, 2003.

[28] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321-357, 2002.

[29] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20-29, 2004.

[30] A. Fernandez, M. J. del Jesus, and F. Herrera, "On the influence of an adaptive inference system in fuzzy rule based classification systems for imbalanced data-sets," *Expert Syst. Appl.*, vol. 36, no. 6, pp. 9805-9812, 2009.

[31] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing*, 2005, pp. 878-887.

[32] H. M. Nguyen, E. W. Cooper, and K. Kamei, "Borderline over-sampling for imbalanced data classification," *Int. J. Knowl. Eng. Soft Data Paradig.*, vol. 3, no. 1, pp. 4-21, 2011.

[33] J. Han, J. Pei, and M. Kamber, "Data mining: concepts and techniques," *Elsevier*, 2011.

[34] S. L. Salzberg, "C4. 5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993," *Mach. Learn.*, vol. 16, no. 3, pp. 235-240, 1994.

[35] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 61, no. 3, pp. 611-622, 1999.

Wacharawan Intayoad earned her BA of Management of Information Systems in 2003 from Thammasat University, Thailand and M.Sc. of Information Systems in 2008 from Lund University, Sweden. Her research interests are data analytics, big data, machine learning, artificial intelligence in education and logistics domains.

Chayapol Kamyod received his Ph.D. in Wireless Communication from the Center of TeleInFrastruktur (CTIF) at Aalborg University (AAU), Denmark. He received M. Eng. in Electrical Engineering from The City College of New York, New York, USA. In addition, he received B.Eng. in Telecommunication Engineering and M. Sci. in Laser Technology and Photonics from Suranaree University of Technology, Nakhon Ratchasima, Thailand. He is currently a lecturer in Computer Engineering program at School of Information Technology, Mae Fah Luang University, Chiang Rai, Thailand. His research interests are resilience and reliability of computer network and system, wireless sensor networks, embedded technology, and IoT applications.

Punnarumol Temdee received B.Eng. in Electronic and Telecommunication Engineering, M. Eng in Electrical Engineering, and Ph.D. in Electrical and Computer Engineering from King Mongkut's University of Technology Thonburi. She is currently a lecturer at School of Information Technology, Mae Fah Luang University, Chiang Rai, Thailand. Her research interests are social network analysis, artificial intelligence, software agent, context-aware computing, and ubiquitous computing.