Probability-Weighted Voting Ensemble Learning for Classification Model

Artitayapron Rojarath and Wararat Songpan
Department of Computer Science Faculty of Science, Khon Kaen University, Thailand
Email: artittayaporn.r@kkumail.com, wararat@kku.ac.th

***Abstract*—Many research studies have investigated ensemble learning. However, these research studies proposed an approach for improving the ensemble learning. We propose the efficiency method using probability weight as a support to the classifier model called the probability-weighted voting ensemble learning, which computes its own probability computation for each model from the training data. This research has tested the proposed model with 5 UCI data sets in various dimensions and generated four models, the 3PW-Ensemble model, the 4PW-Ensemble model, the 5PW-Ensemble model, and the 6PW-Ensemble model. The experimental results of the study yield the highest accuracy. Considering the comparison of efficiency, the accuracy of the proposed model was higher than those of the based classification models and the other ensemble models.**

***Index Terms*—classification model, ensemble learning, machine learning, model combination, probability weight, weight voting**

I. Introduction

The classification model was one of the techniques used for the retrieval of data known as data mining, machine learning and big data. There were actually several techniques for data mining, of which [1] was one of those techniques used to classify the data by finding the model set or the performance set to describe and classify the model types. This paper aims to obtain the model for predicting the data class, and the model was generated by analyzing the training data [2]. Considering the appropriateness of applying the classification model to the data sets, the model that was obtained could be used for determining the data types that had not been categorized in the future, called the unseen data [3]. At present, the classification model has been applied to many kinds of research studies such as medicine, marketing, and geoinformatics [4], [5]. When the data were input to be examined and evaluated for the efficiency through different techniques, the outcome of this procedure was to obtain the classification model with the best prediction and the highest accuracy. In addition, the outcome of the procedure was to reduce the error prediction of the model from analyzing and predicting the data set. According to the above, the various kinds of classification models were tested to evaluate the efficiency of the classification and prediction. Then, those models were evaluated and compared to determine the one with the highest accuracy.

By employing the different classification models to be processed together, pros and cons or strong points were found to be different for each model that was suitable for the different inputs. The limitations of these approaches affected the accuracy obtained from processing each approach. Thus, these factors influenced the efficiency of classification and prediction. The approach with efficiency of classification and prediction for classifying the data types was very important. As a result, the researcher attempted to improve the efficiency of the classification and prediction through the based classification models to determine the results to achieve the aim of enhancing the efficiency of classification and prediction. The data could be either similar or different. The training model, after it was obtained, was used to predict the unknown data [6]. This data analysis was called data prediction through the ensemble learning technique, which is an advanced machine learning technique that could improve the efficiency of prediction. In other words, employing only one approach for determining the best prediction could not possibly result in efficient and stable classification and prediction. Generally, the ensemble learning technique has been employed worldwide, particularly for developing the learning model for the machine learning with a focus on enhancing the efficiency of the model performance. The ensemble learning was the approach that combined different independent models to enhance the efficiency of the model for the classification [7]. As mentioned above, this present study employed the based classification model combined with the ensemble learning to enhance the efficiency of prediction. In addition, the ensemble learning was employed to enhance the efficiency of the model in different study fields. For example, clinical and medical diagnosis employed the ensemble learning to enhance the efficiency of data analysis to generate a supporting system for making decisions for diabetes patients [8]. Moreover, the ensemble learning was utilized to create a new strategy by the combination of various classification models to classify the tumor in the breasts through a mammogram. For text mining, the ensemble learning with the model combination through the ensemble was applied to the classification of the text sentiment. The researcher employed the well-known classification approach for testing in the research studies.

Manuscript received April 20, 2020; revised September 23, 2020.

doi: 10.12720/jait.11.4.217-227

The based classification model should be divergent for increasingly predicting the different data with an approach consisting of a decision tree, Naïve Bayes, K-nearest neighbors, support vector machine, Bayesian network, and multilayer perceptron.

For this study, the ensemble learning employed to combine the models consisted of vote ensemble, bagging, and random forests. Vote ensemble was selected to combine the models. This approach combined the outcomes from the prediction and then selected the most closely corresponding results to determine which result was the most suitable. This approach is called voting [9]. According to the concept of combining the ensemble learning with weight voting in this study, the weight was employed for the vote ensemble to enhance the efficiency of the classification model. Additionally, the weight was raised to improve the prediction of the ensemble model with a focus on adjusting the error prediction result to agree with the actual class. There were many approaches for the ensemble learning to combine the models. For this study, the weight was employed for every class obtained from all models. The weight was selected because the vote ensemble usually had problems when combining the model with the even number. For such a combination, some of the samples could not select the class with the most corresponding answers. Considering the factor of selecting the answer, the model combination through the weight was considered the approach that could select the resulting class with the highest weight as the outcome from the prediction. Many research studies have employed the vote ensemble. For example, the weighted voting ensemble was applied for classifying the text sentiment. Another example was about recognition, which was the combination of the classification model through vote based on the ensemble classifier for recognition [10].

We have structured the paper as follows: In Section II, we describe the related methods. The proposed framework used is discussed in Section III, including data preprocessing, performance evaluation, and a description of the proposed model. The experimental results are explained in Section IV. Finally, our conclusions are presented in Section V.

II. Literature Reviews

A. Based Classification Model

The main objectives of classification were to analyze the input data and to develop an accurate pattern for each feature existing in the data. The well-known-based classification is the information classification. There are several kinds of study fields such as the study of geographic information that apply this information for classifying the data. This approach is used for predicting types of the stones by implementing the data mining and applies the based classification to physical geography. Additionally, accurate identification of the stone type of the carbonate stone in the reservoir could be implemented as the basic procedure for enhancing the performance of constructing the model from industrial features. The performance was analyzed through the based classification with six different algorithms consisting of k-NN, Naïve Bayes, C4.5, random forests, SMO, and multilayer perceptron. According to the experiment, the process of combination before the processing was carried out with the based classification and could result in the accuracy of prediction at the 97.4% level. In addition to the study of the geographic information, the research study of agricultural information is also conducted. The decision tree techniques and dataset were applied to a swine farm as a case study [11]. In the swine farm, the based classification could be used for improving the different detections and weak points for farming management. The decision tree technique was calculated by utilizing the C4.5 algorithms. The intention of this study was to determine the performance of the C4.5 algorithms for classifying the information on swine herds and to identify the weak points of the farm management. The results of the experiment revealed that the decision tree algorithm could detect the relationships and dataset patterns of swine breeding in the village. Moreover, the decision tree algorithm with the decision rule could give the reason for selecting the swine breeders.

The based classification was also implemented with the dataset relevant to other classifications, such as classifying a parking lot with the effective dataset. The locating detection system for an outdoor parking lot drew much interest, since the parking lot was widely utilized. The support vector machine was utilized to detect the location of the parking lot. Generally, constructing the classifiers that were not trained could detect the location of vacant parking lots. The best result from the texture-based classifier was approximately 89%. In another study about commutation, the texture-based classifier detected a traffic accident involving injured people through the decision tree model. Since the trend of traffic is still increasing, this study attempted to encourage more safety on the road. To decrease the number of accidents exactly, it was necessary to recognize the causes of the accident. Thus, the data for traffic accidents were collected and analyzed to avoid accidents in the future. This study aimed to identify the rules resulting from the decision tree algorithm for preventing the traffic accidents for the injured people who had the traffic accident records, causes of the accident, and types of accidents. The result of the study led to the conclusion that the CART algorithm of the decision tree was considered a useful tool for identifying the scene of the accident.

Moreover, the based classification was applied to specify the information and used in the financial transactions of banks or other financial organizations. For example, a genetic algorithm was implemented to enhance the performance of decision in the credit of the banks to avoid the complexity and excessive time in the statistical and mathematic programming. Intelligent techniques were popular for the financial research, particularly for enhancing the performance of decision of the bank. However, the proper decisions for loaning in the credit selection of a bank would possibly increase the interest during a crisis. This study proposed an intelligent model based on a genetic algorithm to cope with the decisions for loaning during credit selection in a bank. Similarly, the based classification was applied in medical diagnosis for the clinical analysis of the input data. According to a case study, the based classification explored the medical diagnosis concerning the analysis of kidney transplantation through the decision tree technique and random forests for predicting the result of kidney transplantation with incompatible antibodies and high risk. The decision tree model and random forests identified the risk factors regarding the acute rejection. In addition, the decision tree model prescribed hazardous identification levels for a specific antibody. This case study marked the potential of discovering new features in the dataset.

In addition to the medical study of diagnosis and treatment, an electronic device was applied to the physical system of the human body, such as the study of an electronic nose (e-nose). Since the ability to identify the odor of human beings was unstable and limited regarding the external influences, this study compared the semi-supervised to supervised algorithms in classifying the e-nose dataset: a case study of tomato juice [12]. The supervised algorithm was the based classification for the data. To examine the e-nose, the based classification needed labeled data that were suitable for training. In this study, the semi-supervised algorithm relied on the Cluster-then-Label with labeled and unlabeled data simultaneously to construct the better classifier with less training data and to start coping with the e-nose data in the first place. The result of the study revealed that the spectral clustering technique based on the semi-supervised algorithm had better performance than the supervised algorithm in all aspects. Additionally, the semi-supervised algorithm could construct a reliable classifier with only a few labels.

B. *Ensemble Classification Model*

In addition to the based classification model, another classification model that is extensively implemented for developing the learning model in the machine learning is called the ensemble learning technique. This technique combines different types of classification to gain the performance of the better prediction [13]. It is applied to various kinds of research studies, particularly the rapid growth of an online social network, such as Facebook or Twitter. Those sentiments have great influence on users [14].

Opinion mining, also known as sentiment analysis, is extensively implemented. There is the research study proposing the new sentiment analysis on the basis of the Text-based Hidden Markov Models (TextHMMs) for classifying sentiments that substitute word consequences in training texts for the sentiment lexicon set beforehand. This sentiment learning model expresses the feelings via the TextHMMs ensemble: it combines TextHMMs together to examine the collocation of words by classifying the feelings of the text input. Additionally, it combines the classifiers with the ensemble-TextHMM by using the bootstrapping technique to gain the final classifier [15].

Ensemble learning is not only applied to the social media but also extensively implemented in the medical affairs in both analysis and diagnosis, for instance, analyzing the data to develop the Clinical Decision Support System (CDSS) for diabetes patients [16]. In this study, ensemble learning developed the CDSS for predicting diabetic retinopathy, revealing that the combination between data preparation and procedures of constructing many models helped improve the performance of the CDSS regarding the data before processing. This algorithm combined the prediction of individual models by calculating the weighted confidence margin of every model. Moreover, there was another interesting study on diagnosis concerning the recurrence of ovarian cancer [17]. The purpose of this study was to investigate the risk factors for the recurrence of the ovarian cancer. For the methodology, the ensemble classification and data mining were combined to prioritize the risk factors and recurrence of ovarian cancer. According to the results of the study, this algorithm prevailed in predicting recurrence of ovarian cancer. In addition to the study of the diagnosis, the mass classification for detecting cancers or tumors was explored [18]. The new architecture was designed to construct ensemble classifiers. This study aimed to examine an easily classified sample with uncomplicated classifiers and data with little flexibility but fewer features to encourage a diversity of strategies such as majority votes, weighted voting, and learning models that could be combinable.

Aside from medical implementation, ensemble learning is also applied to pattern recognition. Ensemble learning is the algorithm that is applied to the Natural Language Processing (NLP) by scoring. The attempt is to explore the most appropriate method of scoring the ensemble learning to enhance the performance of recognition, which is an important procedure of NLP. Voting for searching the class is conducted by binary vote or real vote. The results of every language showed that the ensemble classifier with the real voting technique achieved the goal of enhancing the performance. The ensemble classifier was found to be better than the individual classifier. Many research studies have applied ensemble learning to combine the models or techniques of each topic, including credit scoring and other topics concerning banking and financial transactions [19]. Ensemble learning for determining credit was studied as a case study. Classification through ensemble learning for the credit provided better performance compared to an individual classifier. For example, a study on data mining combined the method of feature selection and ensemble learning for credit scoring [20]. The vital issue relied on the application of combination and occurrence at the same time as the various feature selections and of the ensemble learning classification regarding the parameter value to increase the performance. The proposed mixed model could be applied to credit scoring for clients to determine good clients. To solve the problem of extending credit, an extensive guideline for applying all algorithms at the same time was employed for both classification and feature selection.

III. Research Methodology

A. Based Classification Model

This part presents studies related to improving the efficiency of the classification models in terms of the based classification model and the ensemble classification model. In addition, this part describes the approaches to evaluate the efficiency of the model together with giving examples related to the classification model as follows.

1) Naïve Bayesian learning

The classification model relies on the principle of probability on the basis of Bayes' Theorem and hypotheses that encouraged the independent events. According to the definition, the classification model could generate the classifiers of the Naïve Bayes model as below, where each instance *X* had *N* attributes or *X* = {*A*1,...*An*} and *Ci* as the class label in the equation below.

$$\text{Naive bayesian classifier } = Max(P(C_i)\prod P(A_j|C_i)) \quad (1)$$

From the above equation, the answer was the class, any of V_i results that were selected as the class with the Maximum Probability (MAP class) after being computed. Then, it became the answer to the sample of the required prediction. [21]

2) Bayesian belief networks

Bayesian Belief Networks or so-called Bayes nets were the model of the probability graph. Bayes net intended to reduce the limitation of Naïve Bayes in the hypothesis about the independence between the attributes. This approach described the conditional independence between the variables. The conditional independence referred to the probability of *X* not subject to *Y* when *Z* was $P(X \mid Y, Z) = P(X \mid Z)$.

Each variable of the Bayes net had the proper probability that could be the probability of the initiated node or the probability that was obtained from the relationships of many nodes, which was called joint probability, as in the equation below. [22]

$$\text{Join probability} = P(x_1,...,x_n) = \prod^{n} P(x_i|Parents(x_i)) \quad (2)$$

3) Decision tree

The decision tree involves learning through classifying the data into categories by means of the attributes of the data. This approach was not very complicated. Its structure was likely a tree with branches regarding the conditions or the predicted data. The decision tree needed the conditions for making a decision. To create the decision tree model, the attribute with the closest relationship to the class was selected as the uppermost node of the tree (root node). Then, another attribute was continuously determined. For determining the relationships among the attributes, the information gain was employed. Select the attribute with the highest information gain that could be calculated from the following equation [23] when *S* contains S_i-tuples of class C_i for $i = \{1, \ldots, m\}$

Information measures the information required to classify any arbitrary tuple.

$$\mathrm{I}(s_1,s_2,...,s_m) = -\sum_{i=1}^{m}\frac{s_i}{s}\log_2\frac{s_i}{s} \quad (3)$$

Entropy of attribute A with values $\{a_1, a_2, \ldots, a_v\}$

$$\mathrm{E(A)} = \sum_{j=1}^{v}\frac{s_{1j}+...+s_{mj}}{s}I(s_{1j},...,s_{mj}) \quad (4)$$

Information gained by branching on attribute A:

$$\mathrm{Gain(A)} = \mathrm{I}(s_1,s_2,...,s_m) - \mathrm{E(A)} \quad (5)$$

4) Multilayer perceptron

Multilayer perceptron was one of the artificial neural networks, which was the predicted network typically applied to the task that needed the prediction. It could help specify significance. The multilayer perceptron consisted of multilayer neural networks. Each layer comprised a node, the weight of the node vector (w metric), a bias vector (b) and am output vector (a) when m was the number of the layer rank. Each layer was accepted and calculated the aggregation of inputs and the weight of each node. Then, it transferred those values to another node in the next layer as in the equation below. [24]

$$a^{m+1} = f^{m+1}(W^{m+1}a^m + b^{m+1}) \quad (6)$$

5) Support vector machine

The Support Vector Machine, an approach that could help solve the problem of classification, was employed to classify and analyze the data on the basis of the equation coefficient to create the linear model focusing on the best model. The algorithm used for categorizing the data is shown below. [25]

$$y = sign\left\{\sum_{j=1}^{n} w_j\phi_j(x) + b\right\} \quad (7)$$

where ϕ is the function for changing the nonlinear data to the linear equation that could be classified, and b is bias or threshold.

6) k-Nearest neighbors

k-nearest neighbors was the approach used for classifying the class to determine which class could replace the condition or new cases by determining numbers of cases or conditions (k) that were the same or most nearly the same. The k-NN approach compared the interesting data with other data to see to what extent they were similar. The processing created an answer as if it were the answer for the nearest data. The formula below was used for calculating to determine the nearest distance from (x_1, y_1) to (x_2, y_2) when d was the distance as follows. [26]

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (8)$$

To encourage higher accuracy of the calculation, weighting was employed for each node. k-Neighbors were arranged in the ascending order: $d_1,...,d_k$ was the distance arranged in the ascending order, when d_1 was the least distance for the weight of each node, calculated per the equation below.

$$w_i = \frac{d_k - d_i}{d_k - d_1} \tag{9}$$

B. Ensemble Classification Model

The ensemble method was widely implemented to develop the learning model in the machine learning. This model combined the based classification models to help find the answer, aiming to enhance the effectiveness of classification and outcome properly. Recently, the ensemble classification model recovered findings on combining the different classification models to achieve better performance. The ensemble learning was the model concerning the decision with the strategy of combining the prediction of classifiers to create the new instances [27]. The ensemble method was a new trend of machine learning, which was the method of examining the number of specific classifiers and selecting some of classifiers for constructing the ensemble model. As a result, combining the individual classifiers and predictions achieved better performance than a single individual classifier [28].

Mostly, the research study of classifiers in the ensemble model concerned constructing the ensemble model through the single learning algorithm. For instance, this study implemented the based classification consisting of decision tree, k-nearest neighbors, support vector machines, multilayer perceptron, naïve bayes, and bayesian network from combining base classifiers to construct the ensemble model, called the homogeneous ensemble if the ensemble was constructed by the set of classifiers trained by the same algorithm, whereas it was called the heterogeneous ensemble if the ensemble was constructed by the different algorithms. For this study, it constructed the set of the heterogeneous ensembles. The sets of classifiers with single prediction were commonly combined by majority voting or weighted voting [29]. The key concept behind the ensemble model was valuing the weight of the individual classifier with various methods and combining those classifiers to obtain all classifiers with the better performance.

The important principles of ensemble learning for analysis could be categorized into 6 techniques described as the following.

1) AdaBoost

Adaptive boosting (AdaBoost) was considered a machine learning model that was utilized by combining with other algorithms to improve the better algorithm. AdaBoost was combined with the weak classifier to construct the algorithm as the strong classifier. AdaBoost was an ensemble algorithm that was well-known for improving the simple boosting classification through the iterative process. The principle of boosting (also known as adaptive resampling and combining) was a common method for developing the performance of the weak classifier. For this method, the weak classifier was run repeatedly on the training data with various data distributions. The classifiers constructed by weak classifiers were combined to compose a strong classifier to achieve higher accuracy than a weak classifier. The principle behind the adaptive boosting algorithm relied on the significance of more complicated models. The classification drew the interest in the amount of the weight set in every model of the training set [30], [31]. The AdaBoost algorithm developed the weight of each case by training the set of the same and different weak classifiers by decreasing the distribution and combination of weak classifiers. Last, the AdaBoost algorithm was carried out through weighted majority voting for constructing the final decision.

2) Bagging

Bagging was the short form of the bootstrap aggregating, which was one of the prior methods of the ensemble learning algorithm. Additionally, bagging was one of the easiest and simplest methods for application [32]. Bagging was a well-known technique for the machine learning to reduce the variance without any prediction biases. Bagging was the method of constructing and utilizing multiple predictors to obtain the combined predictors. The variety of the bagging method was derived from the bootstrap model of the training data. The subset of the different training data was randomized and drawn out by substituting from all training data. Additionally, each subset of the training data was utilized for training the different base learners of the same dataset. Individual classifiers were combined by majority voting when the individual classifiers obtained the output from the testing instance. The different outputs derived from the trained classifiers and the majority voting were determined as the final decision [33], [34].

3) Stacking

Stacking was also a well-known ensemble technique to achieve the outcome with the highest accuracy. The structure of stacking was the two-level structure consisting of level-0 (base-level) classifiers and level-1 (meta-level) classifiers. The base-level classifiers were trained by the training dataset and then constructed the prediction. Later, a metaclassifier was trained by metadata to identify the output of the base-level classifier as the class label. Generally, stacking was utilized for combining the developed model with the different classifiers. Those classifiers were combined in different predictions to become the final resolution.

4) Voting

Voting was the simplest method of combing a single classifier algorithm. Selecting a classifier combination was accomplished by analyzing the design of the ensemble classifiers. Voting was the method used to make decisions by selecting only one of several alternatives. Voting depended on the class predicted with the majority voting. Moreover, voting was the most utilized in the ensemble methods. Generally, the voting method consisted of unweighted voting and weighted voting. Unweighted voting consisted of simple voting and majority voting, whereas weighted voting included simple weighted voting [35]. The basic principle of the simple voting and the weighted voting placed the interest on coping with the base classifier with labeled outputs. The examples of voting were the following:

Simple voting was called majority voting and was most extensively applied to the ensemble learning model. Weighted voting was simple voting that was considered the appropriate method for all classifiers with equal performance. However, the base classifier was practically carried out with different weights for defining the weight. Thus, weight voting was designed to define the weight for constructing a strong classifier [36].

5) Random forests

Random forests was the classification model of supervised learning based on the model combination, which was similar to the bagging method [37]. Random forests was similar to the bagging method, but the random forests encouraged the diversity of the model by randomizing the attributes instead of only the sample data. The procedures of random forests combined decision tree prediction with effective aggregation and bootstrap to help examine the regression and two-class and multiclass classification problems. The improved accuracy of the random forests classification resulted from predicting the ensemble method of the tree. After numerous trees were constructed, they were voted on to determine the most popular class. These procedures were called random forests [38], [39].

6) Random subspace

Random subspace was a popular sampling method to enhance the effectiveness of weak classifiers and to improve the accuracy from classifying the individual classifier. Random subspace was the ensemble method where vectors of traditional features with high dimensions were randomized to construct the subspace with the low dimension. Later, multiple classifiers were combined in random subspace for the final decision [40]. For random subspace, the training dataset was improved similarly to the bagging method. This improvement was carried out rather in the feature space than in the instance space. This method benefited from applying random subspace. When constructing the base classifier, the dataset had several complicated or irrelevant features [41]. The results revealed that base classifiers in the random subspace were better than in the traditional feature space.

IV. Proposed Framework

For testing the dataset to determine the accuracy of the data, there were several widely used approaches to evaluate the efficiency of the dataset. One of those approaches was the classification model, which consisted of several models that could test the dataset. Each model was suitable for a test with different datasets. After evaluating the dataset to determine efficiency through classification models, the obtained accuracy was compared to determine the best classification model. This evaluation approach provided the best individual model. Because each model had different efficiencies of evaluation and might have depended on the input management, the models with different classification models were combined to obtain the best classification model. Therefore, ensemble learning was employed to combine the classification models. Likewise, the based classification models were employed to combine the classification models in this study to obtain a classification model that could offer high accuracy and apply to any dataset of the classification data type and any approach to collecting the data.

The classification data type started with data preparation. This procedure prepared the data before evaluating efficiency through different classification models.

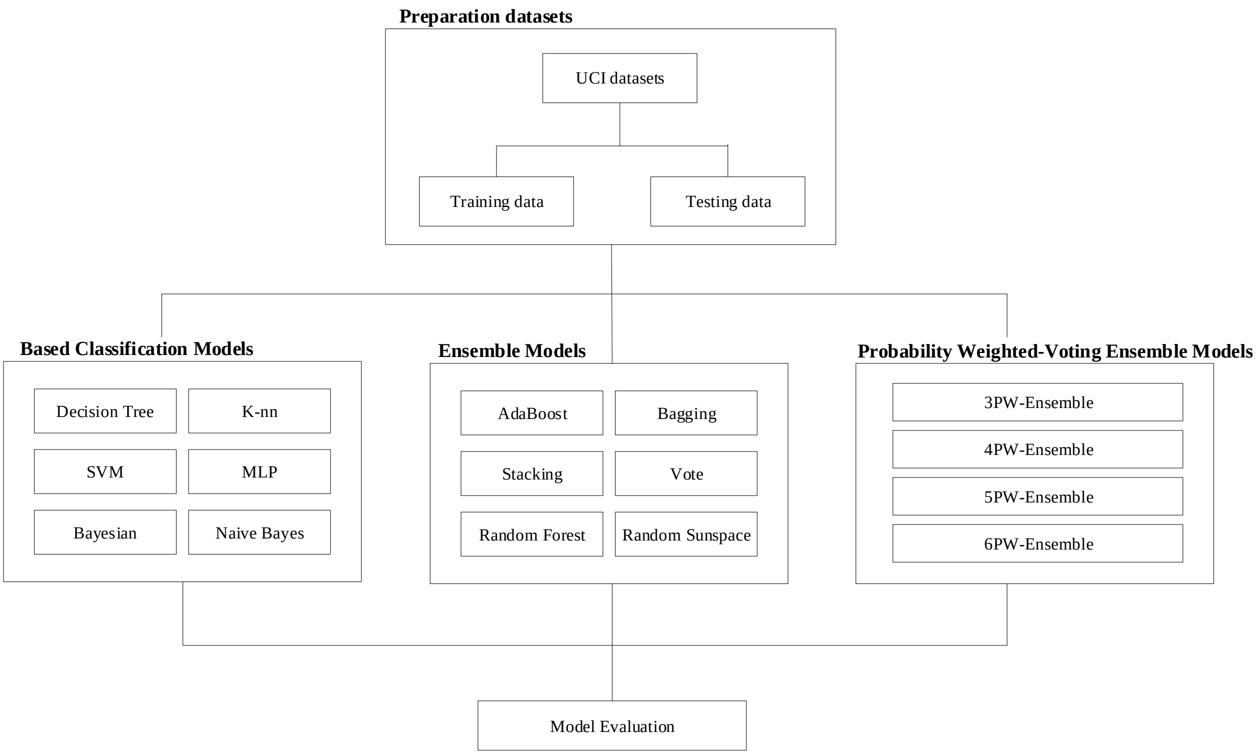

Figure 1. Categories of classification model for testing

The data used for testing were derived from the UCI dataset (Center for Machine Learning and Intelligent Systems), which is a well-known central database for testing the data in the different fields of research studies. In this study, the testing data were divided into two categories: training data for generating the model and testing data for testing the efficiency of the model. The classification model, which is a widely used model for testing, was employed in this study. To test the efficiency of the based classification model and the ensemble classification model as the original ensemble models, these approaches were evaluated together with other developed models, which was called the probability weighted voting ensemble learning. The based classification model was considered the initiated classifier for model improvement. This approach consisted of six main classifications, namely, the decision tree, k-nearest neighbors, support vector machines, multilayer perceptron, naïve Bayes, and Bayesian network. The ensemble classification model also consisted of six techniques, namely, AdaBoost, Bagging, Random Forest, Random Subspace, Stacking, and Vote (as shown in Fig. 1).

Then, the accuracy was evaluated to compare and select the best approach. Fig. 2 shows the overview of probability weighted voting ensemble learning.

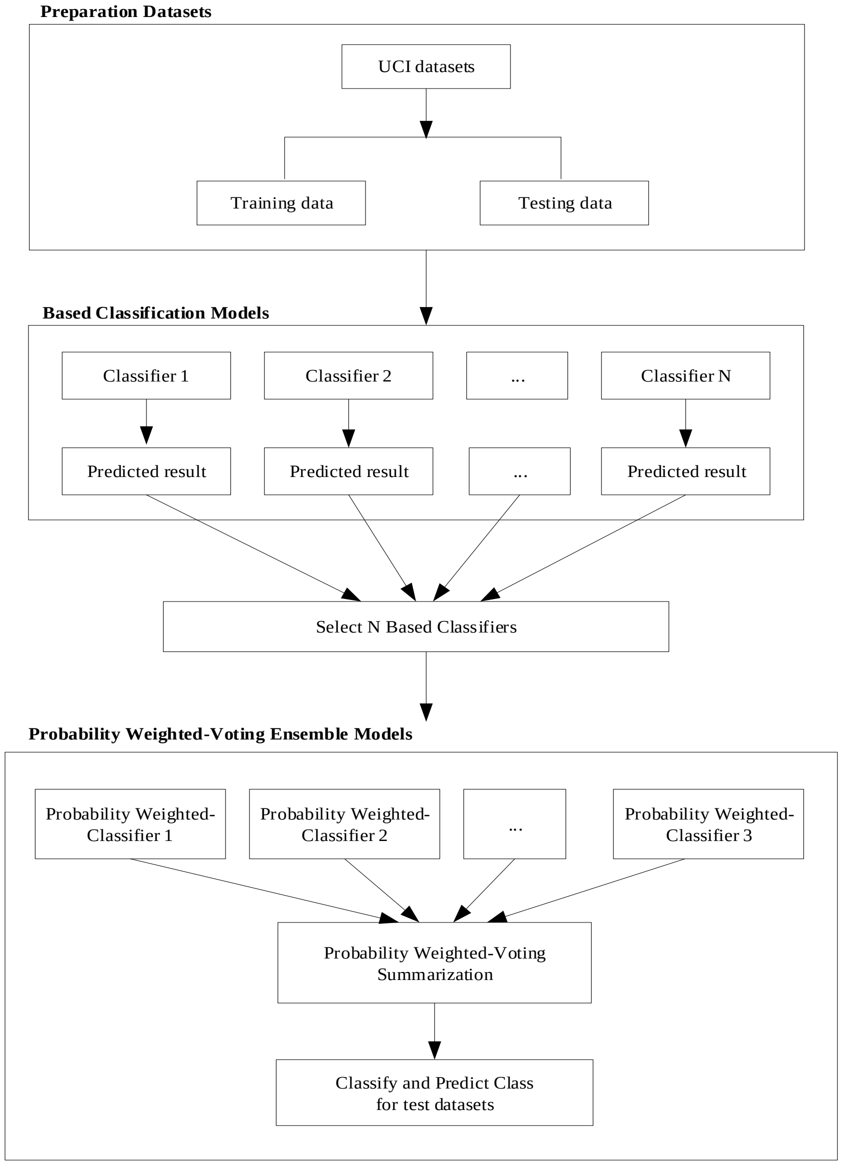

Figure 2. Overview of probability-weighted voting ensemble learning

A. *Overview of Probability-Weighted Voting Ensemble Learning*

Fig. 2 shows an overview of probability-weighted voting ensemble learning, and Algorithm 1 provides a flow chart of the probability-weighted voting ensemble learning. The training dataset was T_a with training samples $T_a = \{(p_1, q_1), (p_2, q_2), \ldots, (p_n, q_n)\}$, while training sample T_a was the instance in the training data. Ta comprised classes of each sample (p, q_n) when n was the number of all samples. The probability-weighted voting ensemble learning first generated the classifier of the base classification. The classifier $m = \{m_1, m_2, \ldots, m_g\}$, and g was the number of all classifier base.

After generating the classifier, the process of selecting the model for combination was carried out. This process used Algorithm 2. The flow chart of this algorithm was applied to the base classification, which was the stage of selecting the classifier for ranking the accuracy in descending order. Moreover, this process was utilized to generate a probability weight obtained from calculating the summation of the initial weights of each classifier and to generate a new ensemble model.

Algorithm 1. Probability weighted voting ensemble learning approach

Required

Input: training dataset T_a and testing dataset T_b

Procedure

1: Base classification generation;
2: Generate classifier $\{m_1, m_2, \ldots, m_g\}$;
3: Call Select model for combine process in Algorithm 2;
4: New ensemble generation;
5: Generate probability weight set $L = (l_1, l_2, \ldots, l_n)$;
6: Generate new ensemble model $\{e_1, e_2, \ldots, e_g\}$;
7: Call probability weighted voting summation ensemble process in Algorithm 3;
8: Probability weight-voting for ensemble result in all samples;
9: Class ensemble results for the samples in T_b;

Output: Probability Weight-Voting Ensemble Learning

The process of the probability-weighted voting ensemble is presented in Algorithm 3. Algorithm 3 was applied to the new ensemble classification for generating the probability weight obtained from the weight summation and every class in set $C = (c_1, c_2,\ldots, c_r)$ of the testing sample set $Y = (y_1, y_2,\ldots, y_n)$ derived from the base classification $m = \{m_1, m_2, \ldots, m_g\}$ to obtain the probability-weighted voting and to become the class ensemble results of samples in T_b and probability-weighted voting $C = \{(w_1m_1, w_1m_2), (w_2m_1, w_2m_2), \ldots, (w_rm_g, w_rm_g)\}$ for the ensemble result.

Finally, the probability-weighted voting obtained from the weight of each classifier in the testing dataset was derived from combining the classifiers. The base classification and the ensemble results were derived from the probability weight that was calculated from the initial weight of the base model through ensemble learning.

B. Selecting N-based Classifiers

Algorithm 2 shows an overview of selecting N-based classifiers. The input of this process was from the base model classification, whereas the obtained output was the new ensemble classification model. This process, which was the first process before generating the probability weight ensemble learning, started with inputting the dataset by dividing it into two sets consisting of the training sample set $X = (x_1, x_2, \ldots, x_n)$ and the testing sample set $Y = (y_1, y_2, \ldots, y_n)$, where n is the number of all samples in the dataset. Each sample comprised the class set $C = (c_1, c_2, \ldots, c_r)$, where r is the number of all classes in the dataset. At this stage, the base model was generated, and the predicted result $p = (p_1, p_2, \ldots p_n)$ was obtained for determining the accuracy, as shown below.

$$\begin{aligned} \%Accuracy &= 100 - \%Error \\ \%Error &= relativeerror * 100 \\ relativeerror &= \left| \frac{x_{mea} - x_t}{x_t} \right| \end{aligned} \tag{10}$$

where x_{mea} is the predicted value of base model, and x_t is the actual value.

After generating the base model and obtaining the accuracy of the classifier, the accuracies of the base model were ranked in the descending order, respectively, to combine the base model to become the new ensemble model.

Algorithm 2. Selecting N-based classifiers

Required

Input: training sample set $X = (x_1, x_2, \ldots, x_n)$;
testing sample set $Y = (y_1, y_2, \ldots, y_n)$;
the class set $C = (c_1, c_2,\ldots, c_r)$;
the accuracies;

Procedure

1: Base model generation;
2: Generate predict result $(p_1, p_2, ..., p_n)$;
3: Calculate the accuracy with accuracies in Eq. (10);
4: Sorted in sequential order based on the accuracy values from max to min;

Output: New ensemble model

Finally, the process of selecting N-based classifiers would select the approach of combining classifiers by means of selecting the classifier with the highest accuracy of all classifiers of the ensemble learning.

Algorithm 3. Probability weighted voting average of ensemble learning process

Required

Input: testing sample set $Y = (y_1, y_2, \ldots, y_n)$;
the class set $a = (a_1, a_2, \ldots, a_r)$;
the probability weight set $L = (l_1, l_2, \ldots, l_n)$
the base classification set $m = (m_1, m_2, \ldots, m_g)$
the probability weighted voting $C = \{(w_1m_1, w_1m_2), (w_2m_1, w_2m_2), \ldots, (w_rm_g, w_rm_g)\}$;

Procedure

1: Initial weights (w) for all the samples;
2: Calculate the probability weighted voting ensemble with probability weight in Eq. (11);
3: Consider each sample (Y);
4: The weight average for each samples (Y);
5: Generate probability weight-voting $(w_1m_1, w_1m_2, \ldots, w_nm_g)$;
6: Generate new class $(w_1c_1, w_2c_2, \ldots, w_nc_r)$

Output: The classes of samples in T_b

C. Selecting N-based Classifiers

Algorithm 3 shows an overview of the probability-weighted voting average ensemble. This process occurred in generating the new probability weight-voting $\{em_1, em_2, \ldots, em_g\}$. The input of this process was the testing sample set $Y = (y_1, y_2, \ldots, y_n)$ in the new ensemble classification model, whereas the output included the classes of the samples in *Tb* when *Tb* was the testing dataset. This process started with calculating the probability weight (l), which was the weight obtained from the base model.

Every testing sample in the testing dataset *Tb* was processed to calculate the probability-weight voting according to $C = \{(w_1m_1, w_1m_2), (w_2m_1, w_2m_2), \ldots, (w_rm_g, w_rm_g)\}$, where C is the weight value derived from combining the models through the ensemble learning of the weight in each class, and g is all models combined in the ensemble learning. The probability-weighted voting could be calculated per the equation below.

$$C = weight \max \sum_{n=1}^{class} \frac{wm_g}{M} \tag{11}$$

C was the class ensemble result obtained from calculating the probability weight that was derived from combining the base model by means of weight summation. Then, the weight summation was divided by $M = \{m_1, m_2, \ldots, m_g\}$. The total weight summation was then calculated to determine the average of weights obtained from all models when M was the base classification that was combined at the stage of the ensemble learning and g was the number of models. The weight was calculated incessantly until the number of $N = Y$, where N is the probability weight in the testing dataset, and Y is the number of the testing sample. At this point, probability-weighted voting = $(w_1m_1, w_1m_2, \ldots, w_nm_g)$ was generated by means of selecting the probability weight with the highest weight, which then became the weight of the class ensemble result when the new class = $(w_1c_1, w_2c_2, \ldots, w_nc_r)$ from each sample in dataset *Tb*.

Finally, the probability weighted voting summation of the ensemble process would calculate the probability weight-voting incessantly until the number of the probability weight in the testing sample was equal to the number of the testing sample set Y. Then, the class ensemble result was calculated to determine the accuracy, which was considered the final result of each sample set.

V. Experimental Result

A. Model Evaluation

There were 5 datasets used as the inputs being tested. These datasets were derived from the UCI dataset, as shown in Table I. The class labels could be binary class and multiclass. Evaluating the efficiency of the based classification model consisted of 6 approaches including the decision tree, k-nearest neighbors, support vector machines, multilayer perceptron, Naïve Bayes, and Bayesian network. Evaluating the efficiency of the original ensemble classification model used 6 approaches including AdaBoost, Bagging, Random Forest, Random Subspace, Stacking, and Vote. In addition to evaluating efficiency through the original ensemble classification model, the new approach to improve the efficiency of the classification model through the ensemble learning was also employed. This approach utilized the weight derived from each based model to combine the models together, which was called the probability-weighted voting ensemble learning. The efficiency testing was divided into 4 approaches comprising the 3PW-Ensemble model, the 4PW-Ensemble model, the 5PW-Ensemble model, and the 6PW-Ensemble model. The efficiencies of the 3 models were tested to compare the accuracies obtained from each model. Then, the accuracies were compared to determine the best approach for each evaluation approach. Then, the best approaches of three models, which consisted of the based classification model, the original ensemble model, and probability-weighted voting ensemble learning, were compared to determine which model offered the highest accuracy and reduced the error of the predicted results. It also determined the model with the highest accuracy.

After testing each model, the efficiency of the classification model was evaluated. At first, the evaluation dealt with evaluating efficiency by processing the based classification model. The testing dataset was input to obtain accuracy as a percentage. Then, the accuracy of each model was compared to determine the best model for those datasets. Another process involved evaluating through the original ensemble model, which provided 6 testing approaches. As considered in Table II, Random Forests offered the highest accuracy of 74.99%, whereas Stacking and Vote both had the lowest accuracies of 53.65%.

Our proposed model was developed from the based classification model by means of ensemble learning with a weight ensemble. This developed approach was embedded in the new ensemble classification called probability-weighted voting ensemble learning.

This approach was created to compare the efficiency of four models consisting of the 3PW-Ensemble model, the 4PW-Ensemble model, the 5PW-Ensemble model, and the 6PW-Ensemble model. According to Table III, the 3PW-Ensemble model offered the highest accuracy of prediction at 85.36%, whereas the 6PW-Ensemble had the lowest accuracy of prediction at 82.84%. Regarding the experimental results, the developed model, probability-weighted voting ensemble learning, was considered the best model when comparing the efficiency of another 2 models, the based classification model and the original ensemble model. The developed model was based on the principle of ensemble learning, which was the classification model.

To evaluate efficiency in this study, the efficiency of the newly developed ensemble learning and original ensemble learning models were compared.

As examined in Table IV, the newly developed ensemble learning model determined that the 3PW-Ensemble model offered the highest accuracy of 85.36%, which was higher than Random Forests using the original ensemble learning model, which had an accuracy of 74.99%. Comparing the developed model to the individual model called the based classification model, probability-weighted voting ensemble learning offered higher accuracy of the classification model.

Therefore, probability-weighted voting ensemble learning with the 3PW-Ensemble model, which was improved by applying ensemble learning to enhance the efficiency of the classification model, could enhance the efficiency of the prediction compared to the accuracy of all classification models.

Thus, according to the analysis of predicted results and accuracy, probability-weighted voting ensemble learning could enhance the efficiency of prediction, reduce the error prediction result, and increase accuracy.

TABLE I. CHARACTERISTIC OF UCI DATA SETS

Dataset	#Instances	#Attributes	#Classes
cpu	70	7	2
b_scale	625	5	3
hepatitis	52	20	2
heart_de	90	14	2
lymp	49	16	4

TABLE II. PERCENTAGE OF ACCURACY VALUES EVALUATED BY THE ORIGINAL ENSEMBLE MODELS

Datasets	AdaBoost	Bagging	Stack	Vote	RF	RS
cpu	**88.57**	80.00	64.29	64.29	**88.50**	**88.57**
b_scale	56.25	74.52	21.63	21.63	**78.37**	63.94
hepatitis	65%	63.46	**65.38**	**65.38**	59.62	**65.38**
heart_de	**77.55**	72.33	57.78	57.78	70.89	73.33
lymp	71.43	**77.55**	59.18	59.18	**77.55**	**77.55**
Average	58.89	73.57	53.65	53.65	**74.99**	73.75

TABLE III. PERCENTAGE OF ACCURACY VALUES EVALUATED BY THE PROBABILITY WEIGHTED-ENSEMBLE

Datasets	3PW	4PW	5PW	6PW
cpu	**91.43**	90.00	90.00	90.00
b_scale	**91.35**	91.26	87.01	81.25
hepatitis	75.00	75.00	**76.92**	75.00
heart_de	**83.33**	84.44	84.44	82.22
lymp	**85.71**	83.67	85.71	85.71
Average	**85.36**	84.87	84.82	82.84

TABLE IV. PERCENTAGE OF ACCURACY VALUES EVALUATED BY THE PROBABILITY-WEIGHTED ENSEMBLE COMPARED WITH ENSEMBLE MODELS

Datasets	3LW	The best accuracy of Ensemble models
cpu	**91.43**	88.57
b_scale	**91.35**	78.37
hepatitis	**75.00**	65.38
heart_de	**83.33**	77.55
lymp	**85.71**	77.55
Average	**85.36**	77.48

VI. CONCLUSIONS

This research study proposed the improvement of the efficiency of classification through ensemble learning to enhance the efficiency of classification and prediction, to obtain better accuracy, and to reduce the error prediction result. In other words, this study encouraged the predicted result to obtain an accuracy that was most likely the actual value of the class. Ensemble learning was employed to develop the model by means of the based classification model consisting of 6 approaches, namely, the decision tree, k-nearest neighbors, support vector machines, multilayer perceptron, naïve Bayes, and Bayesian network.

In addition to the based classification model, the original ensemble model was compared to determine the efficiency of the predicted results using 6 approaches, namely, AdaBoost, Bagging, Random Forests, Random Subspace, Stacking, and Vote. Moreover, probability-weighted voting ensemble learning was also employed, which was a new approach for improving the efficiency of the classification and was divided into 4 approaches consisting of the 3PW-Ensemble model, the 4PW-Ensemble model, the 5PW-Ensemble model, and the 6PW-Ensemble model. This approach considered the ensemble model that combined the ability of each model using ensemble learning by combining the weight occurring in each predicted class. For the classification evaluation metrics, model evaluation was employed by calculating the accuracy that was exposed to the actual class in the percentage. Probability-weighted voting ensemble learning could offer better predicted results. The predicted results could be adjusted increasingly in agreement with the actual class when calculating the probability weight. Thus, it resulted in higher accuracy, as well.

Therefore, this research study proposed the probability-weighted voting ensemble model to improve the efficiency of the prediction to provide suitable efficiency for various kinds of input data. The probability-weighted voting ensemble model combined the based classification model with the weight of each model to obtain new weights of the classes from the predicted results. This approach could offer a model with higher accuracy than the models tested by other approaches. As shown in the three tables displaying the results of evaluating the efficiency of the model, the best approach of the probability-weighted voting ensemble model was the 3PW-Ensemble model, which provided the highest accuracy of 85.36%

CONFLICT OF INTEREST

The authors declare that there are no conflicts of interest regarding the publication of this paper.

AUTHOR CONTRIBUTIONS

Artitayapron Rojarath design and developed the framework of system, and also implementation all models, in charge of bibliographic research, Artitayapron Rojarath and Wararat Songpan analyzed the data, implemented the models, and wrote the paper; all authors approved the final version.

REFERENCES

[1] K. KumarTripathi, "Discrimination prevention with classification and privacy preservation in data mining," *Procedia Computer Science*, vol. 79, pp. 244-253, 2016.

[2] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed., San Francisco, U.S.: Morgan Kaufmann, 2001.

[3] A. Figueira, C. Pitombo, P. D. Oliveira, and A. Larocca, "Identification of rules induced through decision tree algorithm for detection of traffic accidents with victims: A study case from Brazil," *Case Studies on Transport Policy*, vol. 5, no. 2, pp. 200-207, 2017.

[4] P. García-Laencina, J. Sancho-Gómez, A. Figueiras-Vidal, and M. Verleysen, "K nearest neighbours with mutual information for simultaneous classification and missing data imputation," *Neurocomputing*, vol. 72, no. 7-9, pp. 1483-1493, 2009.

[5] J. Shi, S. Zhou, X. Liu, Q. Zhang, M. Lu, and T. Wang, "Stacked deep polynomial network based representation learning for tumor classification with small ultrasound image dataset," *Neurocomputing*, vol. 194, pp. 87-94, 2016.

[6] N. Metawa, M. Hassan, and M. Elhoseny, "Genetic algorithm based model for optimizing bank lending decisions," *Expert Systems with Applications*, vol. 80, pp. 75-82, 2017.

[7] P. D. Almeida, L. Oliveira, A. Britto, E. Silva, and A. Koerich, "PKLot – A robust dataset for parking lot classification," *Expert Systems with Applications*, vol. 42, no. 11, pp. 4937-4949, 2015.

[8] T. Shaikhina, D. Lowe, S. Daga, D. Briggs, R. Higgins, and N. Khovanova, "Decision tree and random forests models for outcome prediction in antibody incompatible kidney transplantation," *Biomedical Signal Processing and Control*, 2017.

[9] E. Santucci, L. Didaci, G. Fumera, and F. Roli, "A parameter randomization approach for constructing classifier ensembles," *Pattern Recognition*, vol. 69, pp. 1-13, 2017.

[10] S. Saha and A. Ekbal, "Combining multiple classifiers using vote based classifier ensemble technique for named entity recognition," *Data & Knowledge Engineering*, vol. 85, pp. 15-39, 2013.

[11] K. Kirchner, K. Tölle, and J. Krieter, "Decision tree technique applied to pig farming datasets," *Livestock Production Science*, vol. 90, no. 2-3, pp. 191-200, 2004.

[12] X. Hong, J. Wang, and G. Qi, "Comparison of semi-supervised and supervised approaches for classification of e-nose datasets: Case studies of tomato juices," *Chemometrics and Intelligent Laboratory Systems*, vol. 146, pp. 457-463, 2015.

[13] A. Onan, S. Korukoğlu, and H. Bulut, "A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification," *Expert Systems with Applications*, vol. 62, pp. 1-16, 2016.

[14] P. Sobkowicz, M. Kaschesky, and G. Bouchard, "Opinion mining in social media: Modeling, simulating, and forecasting political opinions in the web," *Government Information Quarterly*, vol. 29, pp. 470-479, 2012.

[15] M. Kang, J. Ahn, and K. Lee, "Opinion mining using ensemble text hidden Markov models for text classification," *Expert Systems with Applications*, vol. 94, pp. 218-227, 2018.

[16] S. Piri, D. Delen, T. Liu, and H. Zolbanin, "A data analytics approach to building a clinical decision support system for diabetic retinopathy: Developing and deploying a model ensemble," *Decision Support Systems*, vol. 101, pp. 12-27, 2017.

[17] C. Tseng, C. Lu, C. Chang, G. Chen, and C. Cheewakriangkrai, "Integration of data mining classification techniques and ensemble learning to identify risk factors and diagnose ovarian cancer recurrence," *Artificial Intelligence in Medicine*, vol. 78, pp. 47-54, 2017.

[18] O. R. Seryasat and J. Haddadnia, "Evaluation of a new ensemble learning framework for mass classification in mammograms," *Clinical Breast Cancer*, 2017.

[19] J. Abellán and J. Castellano, "A comparative study on base classifiers in ensemble methods for credit scoring," *Expert Systems with Applications*, vol. 73, pp. 1-10, 2017.

[20] F. Koutanaei, H. Sajedi, and M. Khanbabaei, "A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring," *Journal of Retailing and Consumer Services*, vol. 27, pp. 11-23, 2015.

[21] C. Lee, "A gradient approach for value weighted classification learning in naive bayes," *Knowledge-Based Systems*, vol. 85, pp. 71-79, 2015.

[22] Y. Zhao, Y. Chen, K. Tu, and J. Tian, "Learning Bayesian network structures under incremental construction curricula," *Neurocomputing*, vol. 258, pp. 30-40, 2017.

[23] J. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81-106, 1986.

[24] X. Fan, L. Wang, and S. Li, "Predicting chaotic coal prices using a multi-layer perceptron network model," *Resources Policy*, vol. 50, pp. 86-92, 2016.

[25] J. Liu, K. Yan, X. Zhao, and Y. Hu, "Prediction of autogenous shrinkage of concretes by support vector machine," *International Journal of Pavement Research and Technology*, vol. 9, no. 3, pp. 169-177, 2016.

[26] E. Gonçalves, *et al.*, "Prediction of carbonate rock type from NMR responses using data mining techniques," *Journal of Applied Geophysics*, vol. 140, pp. 93-101, 2017.

[27] Y. Chen, M. Wong, and H. Li, "Applying ant colony optimization to configuring stacking ensembles for data mining," *Expert Systems With Applications*, vol. 41, no. 6, pp. 2688-2702, 2014.

[28] A. Onan, S. Korukoğlu, and H. Bulut, "A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification," *Expert Systems with Applications*, vol. 62, pp. 1-16, 2016.

[29] S. Džeroski and B. Ženko, "Is combining classifiers with stacking better than selecting the best one?" *Machine Learning*, vol. 54, no. 3, pp. 255-273, 2004.

[30] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, no. 1-2, pp. 1-39, 2009.

[31] J. Heo and J. Yang, "AdaBoost based bankruptcy forecasting of Korean construction companies," *Applied Soft Computing*, vol. 24, pp. 494-499, 2014.

[32] G. Wang, J. Sun, J. Ma, K. Xu, and J. Gu, "Sentiment classification: The contribution of ensemble learning," *Decision Support Systems*, vol. 57, pp. 77-93, 2014.

[33] F. Petropoulos, R. Hyndman, and C. Bergmeir, "Exploring the sources of uncertainty: Why does bagging for time series forecasting work?" *European Journal of Operational Research*, vol. 268, no. 2, pp. 545-554, 2018.

[34] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.

[35] H. Kim, H. Kim, H. Moon, and H. Ahn, "A weight-adjusted voting algorithm for ensembles of classifiers," *Journal of the Korean Statistical Society*, vol. 40, no. 4, pp. 437-449, 2011.

[36] J. Cao, S. Kwong, R. Wang, X. Li, K. Li, and X. Kong, "Class-specific soft voting based multiple extreme learning machines ensemble," *Neurocomputing*, vol. 149, pp. 275-284, 2015.

[37] J. Abellán, C. Mantas, and J. Castellano, "A random forest approach using imprecise probabilities," *Knowledge-Based Systems*, vol. 134, pp. 72-84, 2017.

[38] R. Genuer, J. Poggi, C. Tuleau-Malot, and N. Villa-Vialaneix, "Random forests for big data," *Big Data Research*, vol. 9, pp. 28-46, 2017.

[39] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 3, pp. 261-277, 2001.

[40] B. Pham, I. Prakash, and D. T. Bui, "Spatial prediction of landslides using a hybrid machine learning approach based on random subspace and classification and regression trees," *Geomorphology*, vol. 303, pp. 256-270, 2018.

[41] G. Wang and J. Ma, "Study of corporate credit risk prediction based on integrating boosting and random subspace," *Expert Systems with Applications*, 2011.

Artitayapron A. Rojarath received her master's degree in Master of Science Program in Information Technology in 2013. Her current research work is focused on data mining and ensemble algorithms. Other researches, she is also interested area of decision support system, information systems and management. She is a Ph.D. candidate in Information Technology program from Khon Kaen University Thailand.

Wararat B. Songpan received her BSc in Computer Science in 2001, and M.Eng in Computer Engineering in 2004, all from Khon Kaen University. In 2010, she received a Ph.D. degree in Computer Engineering from Chulalongkorn University, Thailand. She works with Department of Computer Science at Khon Kaen University. She has been an assistant professor of Computer Science at Khon Kaen Univesrity since 2014. She has been a deputy of head of Computer Science Department from 2014 to present. Her research interests include data mining, machine learning, optimization model, ensemble algorithm, automated software testing and engineering.