*International Journal of Engineering & Technology*, 7 (4.31) (2018) 280-284

International Journal of Engineering & Technology

*Website: www.sciencepubco.com/index.php/IJET*

***Research paper***

# Solving Classification Problem Using Ensemble Binarization Classifier

**M.N. Shah Zainudin[1]*, Md. Nasir Sulaiman[2], Norwati Musapha[3], Thinagaran Perumal[4], Raihani Mohamed[5]**

*[1]Faculty of Computer Science and Information Technology, UPM*
*Serdang, 43400 Serdang, Selangor, Malaysia*
*[2]Faculty of Electronic and Computer Engineering, UTeM*
*Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia*
**Corresponding author E-mail: noorazlan@utem.edu.my*

## Abstract

Binarization strategy is broadly applied in solving various multi-class classification problems. However, the classifier model learning complexity tends to increase when expanding the number of problems into several replicas. One-Versus-All (OVA) is one of the strategies which transforming the ordinal multi-class classification problems into a series of two-class classification problems. The final output from each classifier model is combined in order to produce the final prediction. This binarization strategy has been proven as superior performance in accuracy than ordinal multi-class classifier model. However, learning model complexity (eg. Random Forest-RF ensemble decision trees) tends to increase when employing a large number of trees. Even though a large number of trees might produce a decent accuracy, generating time of the learning model is significantly longer. Hence, self-tuning tree parameter is introduced to tackle this matter. In such circumstances, a number of trees in the RF classifier are defined according to the number of class problem. In this paper, the OVA with self-tuning is evaluated based on parameter initialization in the context of RF ensemble decision tree. At the same time, the performance has also been compared with two classifier models such J48 and boosting for several well-known datasets.

***Keywords***: *Ensemble decision tree; Bagging; Boosting; OVA; PAMAP2; WISDM.*

## 1. Introduction

In many real-life situations, distribution of examples from each class is skewed due to the imbalance data problem. Imbalance data occurs in the event of the representatives of some classes occur much more frequent than other classes [1]. Hence, it is becoming challenging for the learning model to learn the characteristic of each class if the examples have more tendency towards the majority class. The minority class might have lower chance to carry important features for the learning model to deduce the meaningful information. Due to this reason, prediction of two-class classification problem is much easier to be tackled since its class probability is much smaller than multi-class classification problem. In some cases, this condition is considered formidable in order to learn the class relationship and differences from various numbers of attributes. Some classes characteristic might redundant and it is believed that distinguishing one class with another class is also a challenge [2]. Moreover, the traditional multi-class classification model is also incapable to achieve a decent performance when the data sparsely distributed in the input space. Consequently, the learning model is difficult to learn the characteristic of the given attributes with high similarities. For example, in Human Activity Recognition (HAR) paradigm, the traditional learning model unable to produce high accuracy model in predicting the human activity from various sensor position when the similarities in human activity such as walking up, walking down and walking are existed in HAR [3].
Several ways have been reported to overcome the imbalance classification problem. Sampling is one of the former methods to overcome the imbalance class distribution [4]. Undersampling is used to remove some examples from the majority class to balance with minority class. On the other hand, oversampling is applied to increase the example from the minority class. However, it might increase the likelihood of removing the useful information that is considered meaningful for the learning model to perform the knowledge discovery when there is too much information is removed. Yet, overfitting might happen when too many examples are replicated in the second method. Hence, transforming the multi-class classification problem into multiple two-class classification problem introduced to cater this matter. Due to this reason, original multi-class problems are broken down into a series of two-class problems in order to increase the diversity of final class prediction. This method also been known as a class binarization strategy. There are two most prominent strategies could be categorized; One-Versus-All (OVA) and One-Versus-One (OVO) [5]. In OVA, the example is partitioned into several numbers of two-class problems where each problem consists of one class (positive class) and all other classes (negative class) are union with the positive class. In OVO, the example is partitioned into several multiple two-class problems where each problem consists of distinct pairs of class values. In order to obtain the final output prediction, the result from each problem is combined.
The decision tree is essentially known as simplest classifier but yet perform well in solving most classification problems [6][7]. However, in certain cases, this classifier incapable to handle the problem with missing values or containing large number of attributes. Sometimes it might increase the tendency of overfitting when there are too large numbers of growing trees. Hence, an ensemble decision tree becomes a solution to tackle this issue [8]. By com-

bining several numbers of decision trees, it could help to overcome the generalization performance than a single decision tree. In addition, not much research proposed the binarization strategy in the context of ensemble learning model. On the other hand, the complexity of learning model tends to increase when expanding into a number of classification problems using binarization strategy especially when involving a large number of classes. Hence, this research proposed the use of optimal number of trees as a solution to reduce the complexity of the learning model.

In this paper, several contributions are proposed. Firstly, the binarization strategy is introduced in evaluating the prediction performance in the context of an ensemble learning model. Secondly, self-tuning tree parameter is proposed to evaluate the prediction performance using a very minimal number of trees. Thirdly, we also compare the effectiveness of the proposed method with several ensemble models such as boosting and ordinal decision tree, J48. In such states, the proposed method is evaluated with several well-known datasets that are downloaded from UCI Machine Learning Repository as well as WISDM and PAMAP2 that consist of human activity dataset from accelerometer sensor.

## 2. Literature Review

Furkranz introduces binarization strategy for handling multi-class problems in rule learning [5]. Several works have been utilized this learning strategy in solving various classification problems. He and Jin propose OVO on a prediction of human activity in daily living. The Max-Wins voting strategy is applied to predict the next incoming samples. OVO with soft-margins Support Vector Machine (SVM) is reported by Abidine and Fergani in activity recognition [9]. They also compare the proposed work with several state-of-the-art learning models such as Conditional Random Field (CRF) and Linear Discriminant Analysis (LDA). Other work is reported by Martinez et al., which they explored OVA with SVM to classify the slate tile [10]. They claim that OVO produces favorable performance in terms of error rates than OVA. Ng. et al. report an experiment of OVA to solve the problem of simultaneous defect of bearing diagnostic [11]. In order to evaluate the proposed learning strategy, two base classifiers model, namely SVM and C4.5 are utilized. The proposed OVA strategy that is capable to improve the recognition of single defect of diagnosis from two laboratories controlled vibration data set. Wu reports that the adaptive OVA LogicBoost algorithm is able to perform faster in convergence and produce the lowest error rate than an ordinal LogicBoost algorithm [12]. OVA with decision tree is capable to differentiate the attack at the same time able to identify the type of intrusion which has been stated by Gaikwad ad Kulkarni [13]. The proposed combination model has also been compared with the traditional decision tree.

Meanwhile, Varpa et al. report the work on prediction of otoneurological class diseases using OVA and OVO [14]. SVM and K-Nearest Neighbour (KNN) are utilized as a base classifier model. As a result, OVO-KNN is able to produce the highest accuracy result among others. Another work has also been reported by Li et al. They proposed OVA with SVM to recognize the human activity based on traditional handcrafted features [15]. The performance of the proposed method has been compared with several unsupervised learning features such as the sparse autoencoder, denoising auto-encoder, and Principle Component Analysis (PCA). Zhang et al. had proposed OVA strategy with semantic alignments to categorize the visual fine-grained of birds [16].

Several researches are proposed to tackle the categories similarity problem from images. The subcategories are fused and can be learned iteratively using OVA. Another research uses combination of OVA-SVM that also been explored by Wang and Niu. They propose OVA-SVM to solve the problem of intra-inter class imbalanced by maximizing the relative margin of SVM [17]. On the other hand, the exploration of binarization strategy with ensemble learning model has not yet been explored. Adnan and Islam proposed OVA with an ensemble learning model such as random forest [18]. They also compared the performance of OVA-RF with ordinal random forest in terms of average accuracy, the average number of trees and average generation time and experimented on ten well-known datasets.

An ensemble learning works by generating more than one learning models and aggregates the results which are obtained from each learning model. Boosting and bagging are the most prominent ensemble learning model [19]. Both ensemble-learning models are generated based on the theory of decision tree learning. Ravi et al. has compared several ensemble learning models including bagging, boosting, plurality voting, stacking with Ordinary-Decision trees (ODTs) and Stacking with Meta-Decision trees (MDTs) in activity recognition [20]. They claim that plurality voting outperforms than bagging and boosting in terms of average accuracy. Ayu et al. also report the work on comparison classifier performance in defining the human activity[21]. As a result, ensemble-learning models including bagging, boosting and random forest is able to produce decent accuracy. Daghistani and Alshammari explore the work on activity recognition using several ensemble-learning models [22]. The boosting is able to produce higher accuracy with combinational of decision tree such as C4.5. However, there are no enormous differences when OVA are combined with Logistic Regression and Multilayer Perceptron (MLP).

## 3. Materials and Methods

### 3.1. Dataset

In this research, the experiment is conducted on datasets that are downloaded from UCI Machine Learning Repository. Two additional accelerometer human activity datasets are also been utilized, namely WISDM [23] and PAMAP2 [24]. Generally, the dataset consists of multi-classes problem with various types of numbering and nominal attributes. In order to evaluate the final prediction, majority voting is applied to combine the prediction output from each learning model. 10-fold cross validation strategy is used to measure the performance of each learning model. The detail explanation of each data set is illustrated in Table 1.

**Table 1:** Descriptions of datasets

| Dataset name | Attributes | Examples | No of class |
|---|---|---|---|
| **UCI Machine Learning Repository** | | | |
| Balance Scale | 4 | 625 | 3 |
| Car Evaluation | 6 | 1728 | 4 |
| Dermatology | 34 | 366 | 6 |
| Ecoli | 8 | 336 | 8 |
| Glass Identification | 10 | 214 | 7 |
| Hayes-Roth | 5 | 160 | 3 |
| Iris | 4 | 150 | 3 |
| Lenses | 4 | 24 | 3 |
| Soybean (small) | 35 | 47 | 4 |
| Statlog (vehicle) | 18 | 846 | 4 |
| **WISDM** | | | |
| WISDM | 36 | 17100 | 6 |
| **PAMAP2** | | | |
| PAMAP2 - ankle | 36 | 25461 | 17 |
| PAMAP2 - chest | 36 | 25526 | 17 |
| PAMAP2 - wrist | 36 | 25436 | 17 |

### 3.2. Binarization Strategy with OVA

As mentioned, OVA binarization strategy works by transforming the multi-class problems into a series of two-class problems. In each class problem, the particular class representing as a positive class example, while the remaining class is representing as a negative class example. Each training example is used $C$ times, once for each $C$ two-class problems. Fig. 1 shows the example of transforming multi-class problems into multiple two-class problems using OVA strategy.

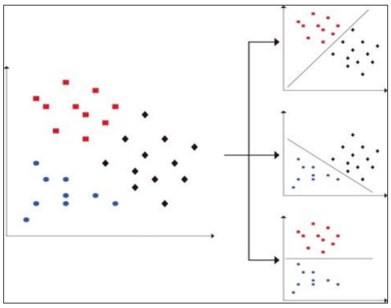

**Fig. 1:** Transforming of multi-class into OVA two-class problems.

Let $S = ((x_1, y_1), \ldots, x_m, y_m)) \in (X \times Y)^m$ becomes the labeled training sample. It will result in $C$-binary classifiers where $h_i: X \rightarrow [-1,1], l \in Y$, each of them looking to discriminate one class $l \in Y$ from all the others. For any $l \in Y$, new re-labeling point in class $l$ with 1 and all others with -1, $h_i$ is obtained by training the binary classifier on the sample set $S$. For each $l \in Y$, assuming that $h_i$ is received from the scoring function $f_l$. Then, the output prediction $h$ of the OVA $h: X \rightarrow Y$ is defined by:

$$\forall x \in X, h(x) = \underset{l \in Y}{argmax} \, f_l(x) \qquad (1)$$

The scores given by the function $f_l$ can be interpreted as confidence scores when $f_l(x)$ is learned as an estimated probability of $x$ conditioned on class $l$. OVA are simple and computational cost obtained from this method is $C$ times of training a binary classification algorithm.

### 3.3. Bagging, Boosting and J48

#### 3.3.1. Bagging

Each model of ensemble generated by a different dataset. Due to its simplicity and good in generalization ability, this method has been applied in solving class imbalance problems [25]. Bagging also able to reduce the data variance by creating the several subsets from training example using random sampling or also known simple bootstrap. A standard decision tree is applied which each node uses the best split among all variables. Unlike boosting, bagging does not depend on the previous decision tree [26]. In order to predict the final output, each output of each model is calculated using majority voting or average prediction strategy.

#### 3.3.2. Boosting

This method iteratively learns the generated model and equal weight is applied to all examples. The weight of incorrectly classified examples is increased from a successive tree on early predictors while decreasing the weight of correctly classified examples [25]. Similarly, with bagging, a majority vote is used for producing the final output.

#### 3.3.3. J48

Decision tree or J48 uses by recursive partitioning the training dataset to observe the possible features for separating the class [27]. Information gain or entropy is used to select the features, which are intuitively deemed to the feature of the lowest entropy (or of the highest information gain). J48 is able to handle numeric attributes, continuous data, and missing values, able to use the attributes with different weights and also can create the pruned tree after the tree being created.

### 3.4. Self-Tuning Random Forest Tree Parameter

Graphs and other numbered figures should appear throughout the text as close to their mention as possible. Figures shouldn't infringe upon the page borders. Random forest (RF) is introduced by adding randomness layer on the bagging classifier method [8]. The n-trees are randomly created, and each example is predicted using each created decision tree. The final class is predicted based on the class that received the highest vote as visualized in Fig. 2. Unlike bagging, each node in a random forest is split using best randomly chosen node among the subsets of a predictor. This method also could avoid the model becomes overfitting. In such states, a large number of trees are able to produce better in prediction estimation.

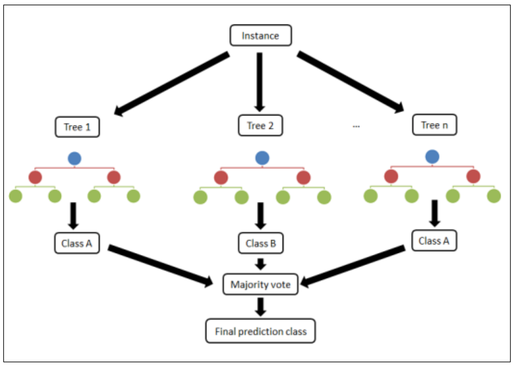


**Fig. 2:** Random forest ensemble model.

Meanwhile, the complexity and cost of collecting a larger sample will be higher when handling with a large number of trees. Hence, the use of less number of trees might consider as an option to reduce the learning model complexity. The number of $n -$tree should simply be set to the largest computationally manageable value or whether a smaller $n -$tree might in some cases be better, in this case $n -$tree should ideally be tuned carefully [28]. In this research, the proposed self-tuning parameter is to define the number of trees in a RF. As starting point, a number of trees are initialized according to the number of class problem. Let $T$ is the number of generating tree, $C$ is a number of class problem and $S$ represented as training examples, where $m$ is the number of two-class problems; $m = 1 \ldots C$. Hence, a number of trees are generated is $Tm = C$.

## 4. Experimental Results and Discussions

This section will present two parts of experimental results and discussion. First part is measured for the UCI Machine Learning Repository dataset, while the second part is for WISDM and PAMAP2. For each dataset, the performance is evaluated based on the learning model using several numbers of trees. This experiment chooses the maximum number of trees, 100 and $C$ is denoted as the number of class. Three performance indicators are applied to measure the experimental result, such as average accuracy, numbers of trees and building time of generating the trees in average 10 times training validation.

### 4.1. UCI Machine Learning Repository

For clear visualization, two groups of the experiment are separated based on the number of distinct class values. Fig. 3(a) and (b) shows the average accuracy obtained from each dataset using the different number of trees. The first group is conducted for the dataset that consists of less than 6 classes is illustrated in Fig. 3(a). Meanwhile, the dataset that are containing more than 6 classes are conducted in the second group in Fig. 3(b).

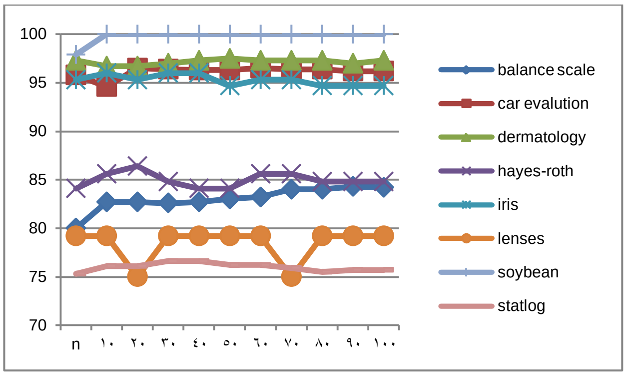


**Fig. 3 (a):** Average accuracy of different numbers of trees (first group).

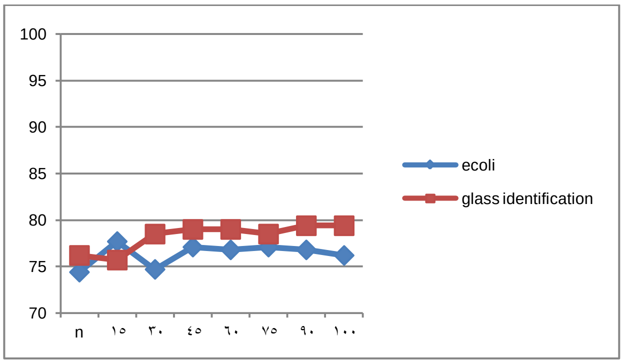


**Fig. 3 (b):** Average accuracy of different numbers of trees (second group).

Obviously, there are no enormous differences in terms of average accuracy of different number of trees for both groups. Dermatology and Lenses are able to achieve higher accuracy when $n$ is 6 and 3 accordingly. The accuracy of Balance Scale and Glass Identification are slightly increasing when the number of trees increases. Meanwhile, the accuracy of Iris and Statlog are decreasing when above 40 of trees has applied. However, the accuracy of the Soybean remains stagnant even though the number of tree has increased. Table 2 shows the number of trees required to achieve an optimal accuracy for each learning model. Consequently, the average number of trees generated from OVA-RF generates reasonably lesser tree than J48 and boosting. An average 21 trees are necessary to create the decision boundary using OVA-RF that can be considered as less as compare with J48 and boosting. Less than 10 trees are required to produce an optimal accuracy for Dermatology and Lenses.

**Table 2:** Number of trees – UCI Machine Learning Repository.

| Dataset | J48 | Boosting | OVA-RF |
|---|---|---|---|
| Balance Scale | 103 | 145 | 70 |
| Car Evaluation | 120 | 188 | 20 |
| Dermatology | 15 | 15 | 6 |
| Ecoli | 43 | 351 | 15 |
| Glass Identification | 59 | 37 | 45 |
| Hayes-Roth | 21 | 33 | 20 |
| Iris | 9 | 15 | 10 |
| Lenses | 7 | 7 | 3 |
| Soybean (small) | 7 | 7 | 10 |
| Statlog (vehicle) | 195 | 159 | 10 |
| Average | 58 | 96 | 21 |

Hence, it could be concluded that the OVA-RF more preferable by minimizing the complexity of generating fewer number of trees which leads to minimizing the learning complexity. On the other hand, this experiment also measures the time taken to build the learning model. Table 3 shows the overall time required for building the learning model.

**Table 3:** Overall time of learning model (in seconds).

| Data set name | J48 | Boosting | OVA-RF |
|---|---|---|---|
| Balance scale | 0.01 | 0.29 | 0.01 |
| Car evaluation | 0.00 | 0.16 | 0.01 |
| Dermatology | 0.01 | 0.06 | 0.01 |
| Ecoli | 0.00 | 0.05 | 0.02 |
| Glass identification | 0.00 | 0.15 | 0.01 |
| Hayes-Roth | 0.00 | 0.02 | 0.00 |
| Iris | 0.00 | 0.01 | 0.00 |
| Lenses | 0.00 | 0.00 | 0.00 |
| Soybean (small) | 0.00 | 0.00 | 0.00 |
| Statlog (vehicle) | 0.02 | 0.23 | 0.03 |
| Average | 0.004 | 0.097 | 0.009 |

The average time taken (in seconds) of the proposed OVA-RF with 0.009s is defeated by J48 with 0.004s. However, boosting obtains 0.097s recorded as the longest learning time for building the tree structure. Furthermore, this research also compare the model performance of the proposed method with reported work by [18]. The authors also explore the use of OVA in the context of RF classifier. Table 4 shows the comparative performance in terms of accuracy, number of trees and generating time (in seconds).

**Table 4:** Comparison with previous work.

| | Adnan and Islam [15] | | | Proposed OVA-RF | | |
|---|---|---|---|---|---|---|
| Dataset name | Acc | Tree | Time | Acc | Tree | Time |
| Balance Scale | **0.826** | 66 | 0.409 | 0.800 | 3 | 0.01 |
| Car Evaluation | 0.817 | 42 | 0.291 | **0.958** | 4 | 0.01 |
| Dermatology | 0.927 | 10 | 0.153 | **0.973** | 6 | 0.01 |
| Ecoli | **0.845** | 10 | 0.110 | 0.744 | 8 | 0.02 |
| Glass Identification | 0.736 | 13 | 0.223 | **0.762** | 6 | 0.01 |
| Hayes-Roth | 0.703 | 14 | 0.107 | **0.841** | 3 | 0.00 |
| Iris | 0.947 | 7 | 0.044 | **0.953** | 3 | 0.00 |
| Lenses | 0.783 | 5 | 0.032 | **0.792** | 3 | 0.00 |
| Soybean | **1.000** | 5 | 0.039 | 0.979 | 4 | 0.00 |
| Statlog | 0.735 | 65 | 1.563 | **0.753** | 4 | 0.03 |
| Average | 0.832 | 24 | 0.297 | 0.856 | 4.4 | 0.009 |

Overall, the results of the proposed method OVA-RF obtains an average of 85.6% accuracy that can be considered higher from the previous work [15]. The proposed method is able to produce an improvement in accuracy for 7 different datasets of UCI Machine Learning Repository. In addition, this method successfully obtains a minimum number of trees that are about 16% less from the previous work at the same time is able to achieve an acceptable performance. On the other hand, the number of attributes and examples are significantly associated with the chosen number of trees. For instance, it might insufficient to produce good performance for learning model to learn the pattern with very less number of attributes using a very minimal number of trees for balance scale dataset. Otherwise, less number of trees is believed as adequate to imply the rule when handling a high variety of attributes and examples. Moreover, few numbers of trees are also believed as necessary to produce an optimal performance for the problem that consists of more examples such as Statlog.

### 4.2. WISDM and PAMAP2

These datasets are selected as an additional experiment using accelerometer sensor data on human activity datasets. In such circumstances, this research chooses 6 trees for WISDM and 17 trees for PAMAP2 according to its class respectively. The detail explanation on the preprocessing and feature extraction can be referred to [29]. Table 5 presents the comparative performance of WISDM and PAMAP2 in the respect of average accuracy and number of trees.

**Table 5:** Comparison performance of WISDM and PAMAP2.

| Data set name | J48 | Boosting | OVA-RF |
|---|---|---|---|
| **WISDM (6 class)** | | | |
| Accuracy | 0.990 | **0.998** | **0.998** |
| Number of trees | 335 | 323 | 6 |
| | | | |
| **PAMAP2 (17 class)** | | | |
| **Ankle** | | | |
| Accuracy | 0.962 | **0.996** | **0.996** |
| Number of trees | 1261 | 799 | 17 |
| **Chest** | | | |
| Accuracy | 0.963 | **0.996** | **0.996** |
| Number of trees | 1159 | 943 | 17 |
| **Wrist** | | | |
| Accuracy | 0.968 | **0.998** | **0.998** |
| Number of trees | 1165 | 873 | 17 |

From Table V, the accuracy obtained from OVA-RF and boosting for both WISDM and PAMAP2 are similar. Though, J48's accuracy is lower compared with the proposed OVA-RF and boosting on both datasets. Meanwhile, the number of trees generated by J48 and boosting somewhat comparable to WISDM is about 300 trees to obtain the optimal performance. Moreover, J48 produces very large number of trees particularly when it is involving with high distinct number of classes (17 class) such as PAMAP2. However, the number of trees generated from boosting showed slightly lesser for each sensor position; ankle, chest and wrist. As expected, less number of trees is attained from OVA-RF at the same time can produce a convincing performance model.

## 5. Conclusion

In this article, the proposed method emphasizes on the work regarding an evaluation of binarization strategy using OVA with the context of an ensemble decision tree. In order to evaluate the performance of the proposed work, OVA-RF is tested using several datasets. From this work, OVA demonstrate the capability to prove its effectiveness in terms of accuracy model and the optimal number of generating trees. In addition, it can be shown a good potential in the context of an ensemble classifier model such as RF. Even though there are no enormous differences in accuracy than ordinal decision tree such J48 and boosting, OVA-RF is believed could minimize the learning complexity by using an optimal number of trees. The research also evaluates the performance of OVA-RF with several additional datasets recorded from an accelerometer sensor. The experimental results also indicate a high accuracy performance in predicting the human activity from various sensor positions that always suffer with similarities of activity type. Hence, it can be concluded that by proposing self-tuning tree parameter in RF significantly could minimize the learning complexity that yields to improve the classification accuracy. In the future, this research can be expanded into applying the OVA binarization strategy on some other learning models such as SVM or MLP. The other binarization strategies including OVO also can be explored in the future works.

## Acknowledgement

This is work supported by University Putra Malaysia research grant.

## References

[1] Sun Z., Song Q., Zhu X., Sun H., Xu B., and Zhou Y. (2015), "A novel ensemble method for classifying imbalanced data," *Pattern Recognit.*, vol. 48, no. 5, 1623–1637.

[2] Chandrashekar G. and Sahin F. (2014), "A survey on feature selection methods," *Comput. Electr. Eng.*, vol. 40, no. 1, 16–28.

[3] Mohamed R., Zainudin M. N. S., Sulaiman N., and Perumal T. (2018), "Multi-label Classification for Physical Recognition from Various Accelerometer Sensor Positions," *J. Inf. Commun. Technol.*, vol. No. 2, no. 18, 209–231.

[4] Krawczyk B. (2016), "Learning from imbalanced data : open challenges and future directions," *Prog. Artif. Intell.*.

[5] Fürnkranz J. (2002), "Pairwise Classification as an Ensemble Technique," *Mach. Learn. ECML 2002*, vol. 2430, no. 2000, 9–38.

[6] Quinlan J. R. (1986), "Induction of Decision Trees," *Mach. Learn.*, vol. 1, no. 1, 81–106.

[7] Mohamed R., Perumal T., Sulaiman N., Mustapha N., and Abd Manaf S. (2017), "Tracking and Recognizing the Activity of Multi Resident in Smart Home Environments," *J. Telecommun. Electron. Comput. Eng.*, vol. 9, no. 2–11, 39–43.

[8] Breiman L. (2001), "Random forests," *Mach. Learn.*, vol. 45, no. 1, 5–32.

[9] Abidine M. B.and Fergani B. (2012), "Evaluating C -SVM , CRF and LDA Classification for Daily Activity Recognition,".

[10] Martínez J., Iglesias C., Matías J. M., Taboada J., and Araújo M. (2014), "Solving the slate tile classification problem using a DAGSVM multiclassification algorithm based on SVM binary classifiers with a one-versus-all approach," *Appl. Math. Comput.*, vol. 230, 464–472.

[11] Ng S. S. Y., Tse P. W., and Tsui K. L. (2014), "A One-Versus-All Class Binarization Strategy for Bearing Diagnostics of Concurrent Defects," *sensors*, vol. 14, 1295–1321.

[12] Wu K. (2015), "AOSA-LogistBoost : Adaptive One-Vs-All LogistBoost for Multi-Class Classification Problems," in *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, 654–662.

[13] Gaikwad V. S. and Kulkarni P. J. (2012), "One Versus All classification in Network Intrusion detection using Decision Tree," *Int. J. Sci. Res. Publ.*, vol. 2, no. 3, 1–5.

[14] Varpa K., Joutsijoki H., Iltanen K., and Juhola M. (2011), "Applying One-vs-One and One-vs-All Classifiers in k -Nearest Neighbour Method and Support Vector Machines to an Otoneurological Multi-Class Problem," *User Centred Networked Heal. Care*, 579–583.

[15] Li Y., Shi D., Ding B., and Liu D. (2014), "Unsupervised learning for human activity recognition using smartphone sensors," *Expert Syst. Appl.*, vol. 41, 6067–6074.

[16] Zhang X., Xiong H., Zhou W., and Tian Q. (2016), "Fused One-vs-All Features With Semantic Alignments for Fine-Grained Visual Categorization," *IEEE Trans. IMAGE Process.*, vol. 25, no. 2, 878–892.

[17] Wang X. and Niu Y. (2013), "New one-versus-all ν -SVM solving intra – inter class imbalance with extended manifold regularization and localized relative maximum margin," *Neurocomputing*, vol. 115, 106–121.

[18] Adnan M. N. and Islam M. Z. (2015), "One-Vs-All Binarization Technique in the Context of Random Forest," in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.*, 2015, no. April, 22–24.

[19] Bolon-Canedo V., Sanchez-Marono N. and Alonso-Betanzos A. (2005), "Data classification using an ensemble of filters," *Neurocomputing*, vol. 135, 13–20.

[20] Ravi N., Dandekar N., Mysore P., and Littman M. L. (2005), "Activity Recognition from Accelerometer Data," *Am. Assoc. Artif. Intell.*, 1541–1546.

[21] Ayu M. A., Ismail S. A., Abdul Matin A. F., and Mantoro T. (2012), "A comparison study of classifier algorithms for mobile-phone's accelerometer based activity recognition," *Procedia Eng.*, vol. 41, no. Iris, 224–229.

[22] Daghistani T. and Alshammari R. (2016), "Improving Accelerometer-Based Activity Recognition by Using Ensemble of Classifiers," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 5, 128–133.

[23] Kwapisz J. R., Weiss G. M., and Moore S. a. (2011), "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explor. Newsl.*, vol. 12, 74.

[24] Reiss A. and Stricker D., "Introducing a new benchmarked dataset for activity monitoring," in *Proceedings - International Symposium on Wearable Computers, ISWC*, 2012, 108–109.

[25] Bhatt J. (2014), "A Survey on One Class Classification using Ensembles Method," *Int. J. Innov. Res. Sci. Technol.*, vol. 1, no. 7, 19–23.

[26] Zainudin M. N. S., Sulaiman N., Mustapha N., and Perumal T. (2015), "Activity Recognition based on Accelerometer Sensor using Combinational Classifiers," in *ICOS2015*, 68–73.

[27] Hall M. a. and Smith L. a. (1998), "Practical feature subset selection for machine learning," *Comput. Sci.*, vol. 98, 181–191.

[28] Probst P. and Boulesteix A.-L. (2017), "To tune or not to tune the number of trees in random forest?," no. 2001, 1–20.

[29] Zainudin M. N. S., Sulaiman N., Mustapha N., and Perumal T. (2017), "Pairwise Classification Using Combination of Statistical Descriptors with Spectral Analysis Features for Recognizing Walking Activities," *J. Telecommun. Electron. Comput. Eng.*, vol. 9, no. 2, 55–60.