Int. J. Advance Soft Compu. Appl, Vol. 7, No. 3, November 2015
ISSN 2074-8523

Comparison of Diverse Ensemble Neural Network for Large Data Classification

Mumtazimah Mohamad[1]**, and Md Yazid Mohd Saman**[2]

[1]Faculty of Informatics and Computing,
Universiti Sultan Zainal Abidin, Terengganu, Malaysia
e-mail: mumtaz@unisza.edu.my
[2]School of Informatics and Applied Mathematics,
Universiti Malaysia Terengganu, Malaysia
e-mail: yazid@umt.edu.my

Abstract

In a large dataset classification, a higher number of attributes commonly evolve over time, where many dynamic learning strategies have been proposed such as the ensemble network and incremental neural network. Ensemble network is a learning paradigm where many neural networks are jointly used to solve a problem. The relationship between the ensemble and component of neural networks is analyzed from the context of classification in integrated framework. This task would reveal that, it may be better to have many neural networks instead the incremental neural network. Most approaches of ensemble using totally different classifiers for prediction. Then, in order to find an appropriate neural network from ensemble members, it can be selected from a set of different available neural networks. Thus, a Distributed Reordering Technique (DRT) is proposed. DRT is an enhanced algorithm based on distributed random for different neural networks. The weights are randomly assigned to networks in order to evolve, so that they can characterize each neural network to some extent of fitness in constituting a better result. The ensemble network integrated framework supported by the selection of some neural networks based on output and weights that made up the ensemble. The experimental study shows that in comparing with some ensemble approaches such as Bagging, DRT can generate a neural network with enhanced performance and stronger generalization ability. Furthermore, the use of DRT for neural network classifier is practical and relevance to classification systems for large and can be applied to different large data dimension in future.

Keywords: *Ensemble Network, Incremental Learning, Large Data, Neural Network.*

1 Introduction

The term 'large data' can refer to different kinds of problems with a large number of features, samples or categories [1]. A large number of dataset can cause the training procedure to be unfeasible for many types of classifiers. The applications such as speech or handwritten character recognition or other problems containing thousands of different classes still remain a challenge [2]. The large datasets classification using machine learning is rarely being discussed by previous research. Ciresan et al. (2010) has showed the detail work on a large dataset for classification tasks using one technique of Artificial Neural Networks (ANN) that imposes on graphical processing unit instead of central processing unit. ANN provides a supervised learning algorithm that performs fine-granule local optimization. In addition, ANN has the ability to learn complex nonlinear input-output relationships by a sequential procedure and adapting themselves [3]. ANN offers the best learning approach besides template matching, statistical classification, and syntactical matching [4].

The use of a single ANN usually leads to the unstable learner and it is sensitive to the initial conditions. However, it works differently for different training data [5]. Previous researchers have proved that ensemble networks can outperform their base of ANN model since individual ANN tends to make errors on different examples [6]. Therefore, to preserve the capability of ANN, an aggregation technique has to be employed. Kittler et al. (1998) mentioned the necessity for a theoretical framework to describe the combinations of classifiers and proposed an output combine strategy which is called the ensemble aggregation. The outputs from multiple ANN ensemble networks are required to be in diverse conditions [7]. The outputs from multiple ANN ensemble networks are required to be in diverse conditions [8]. It is because when different areas of input spaces have been learned by some classifiers, the classifiers become an expert in a particular area of the input spaces, and consequently have fewer errors in those areas. Furthermore, the architecture of neural networks itself is determined by trial and error, and it is not unique. Integrating different neural network using output aggregation strategy is an effective way to solve the variety of output network, and it is rather easy [9]. It only makes sense only if the classifiers are diverse or in other word statistically independent.

This paper proposed the techniques that embedded in several strategies for a large dataset by having clusters of ANN classifiers that work independently to allow the classification task. The reordering technique associated with particular input space is embedded to the classifiers in creating a significant identity. The classifiers output will impose a selection and aggregation process to determine a solid and better ensemble output.

2 Related Works

The approaches of ensemble system can be based on the problem domain instead of improving the performance of classifier generalization. Ensemble based systems can be used in problem domains rather than improving the generalization performance of a classifier. They are incremental learning, error-correcting output code, and feature selection. Incremental learning refers to the ability of an algorithm to learn from a new data that may become available after the classifier has generated previously from available dataset [10]. A previous research [11] has modelled the incremental learning for subsets of larger problem of base classifier to be included in the ensemble. They found that the incremental ensemble had statistically significantly higher accuracy than bagging and random subspace methods. It might need additional information from new batches of data incrementally while preserving previously acquired knowledge [12]. The concept of combining classifiers explores a new direction for the improvement of the classification accuracy. Another previous research [13], has demonstrated the performance of online learning for ensemble incremental learning with three classifiers with ensemble voting method. Some previous studies take into account an ensemble incremental learning that support the ANN. This is because the choice of selecting the base classifier is related to the ability of generalization for the classification tasks of large datasets. The heterogeneous ensemble of ANN model is expected to improve the accuracy as well as the scalability of the datasets.

To develop a cluster of classifiers rather than single classifier, the diversity of classifiers error is a must as well as the selection and combination of the ensemble method. These criteria are the indicator of improved and useful classifiers. The generalization ability of ANN classifier will be utilized in order to get the best accuracy score by using different techniques. It is an alternative approach for improving the result of a classifier of large datasets [14]. In order to get high classification performance from difference classifier, there are several important requirements for both ensemble classifiers and ensemble strategies [10]. Firstly, each individual network should have enough training data and each of the members of the ensemble must have a complementary set of classifier [15]. Secondly , the number of training for each pattern and the network size are the two important factors in measuring the performance of neural networks [16].

The process of designing an ensemble strategy consists of two main steps. The first step is related to the diversity of ensemble where each individual of ANN is trained according to an alternative design specification. In this case, the diversity of ensemble is generated using neural network with the diversity strategy [17]. The difference between classifiers is interpreted as diversity or making errors on different examples in parallel classifiers of learning [18]. A classifier that is diverse would have the ability to find the extent of diversity among the classifiers and estimate the improvement or deterioration inaccuracy of individual classifiers when they have been aggregated [19]. An aggregation of a set of identical ANN resulted in similar generalization abilities in the same way and would neither contribute to the diverse feature nor improve the significant difference between classifiers accuracy [20]. Therefore, in order to be effective, individual experts must exhibit some level of diversities among themselves [21]. The second step is the determination of the suitable aggregator or combiner, which consists of selecting the most accurate aggregator for the generated ensemble [7]. An aggregator can be developed based on the modification of the training algorithm technique-based or based on the modification of learning set technique-based. The output of the ensemble of ANN is usually accurate than any independent network output [12]. It is because the independent network within the ensemble network can potentially have different weight and methods in creating the diversity of the networks. Common approaches of high-resolution representations which use cluster solution, the ensemble will have n (the number of examples) in its time and memory complexity approximation [5].

3 Proposed Methods

The proposed framework were data preprocessing, reordering technique for data resampling, individual ANN classifier learning, selection of classifiers, and aggregation of ensemble networks. As shown in Fig. 1, initially, the dataset were pre-processed and divided into training set, testing set, and validation set. The training dataset applied the reordering technique for data resampling. The result of the reordering was configured with the partitioning task. Each task for the individual ANN was passed on to the corresponding main classifier at the input. The output was transformed into reliability value and then passed back to the main classifier for an aggregated ensemble strategy. In constructing this kind of method, there were five phases in the learning framework proposed for the parallel classification process.

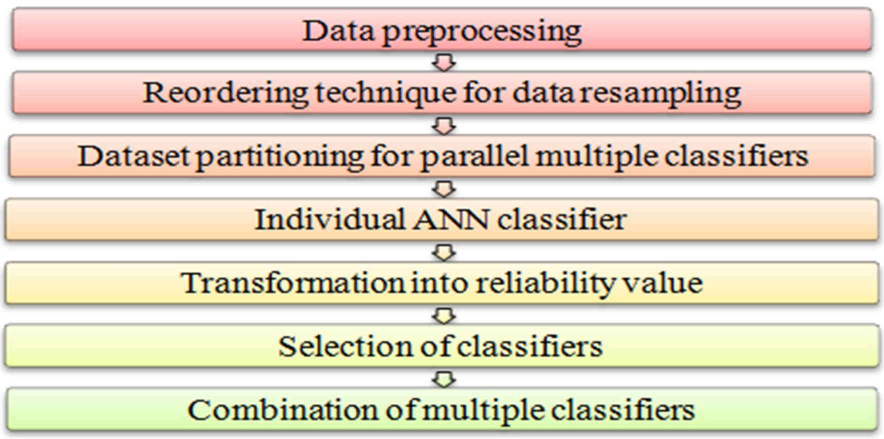

Fig.1: Proposed ensemble framework

3.1 Data preprocessing

MNIST was selected and divided into 60000 examples as the training set and 10000 examples for the testing set. The validation dataset was drawn from the training set which was 10000 examples, and the training set became 50000 examples. The training set were normalized based on (1),

$$f(x) = \frac{maximumValue - minimumValue}{N} \quad (1)$$

where the $maximumValue$ and the $minimumValue$ are the largest and the smallest value in specific range value, respectively; and N is the count of the value. This method was adapted from a previous study [22] that agreed with a previous research regarding data sampling. The input of MNIST was the attribute of the row which included 786 column attributes of image pixels. This normalization was applied to the network input. The other case also utilized the normalization using tanh (hyperbolic tangent) function to return the value in [-1,1] as in (1).

3.2 Reordering techniques

Reordering procedure was the main tasks for proposed ANN cluster in order to create a diversity and maximize the differences between the clusters of ANN [23]. The individual network in each classifier was desperately required to use the batch learning mechanism with respect to scalability of large dataset and it has been examined in the previous chapter. Furthermore, maintaining the original sequence can let all ANN fall in the same or very similar configuration and the training condition was very low. The original ordering was in sequence order at the beginning of the learning process by keeping the initial order fixed in which will generate $N_{network}$, training data T, and validation set V. This network was impractical for the ensemble of ANN because there would be no improvement to the classifiers if the involved training data is small and similar to other network in parallel classifiers [24-26].

A bagging sampling algorithm was used to ensure different samples with different training data subsets that were selected from the original datasets ordering [27]. It efficiently constructs a reasonable size of training data from total N examples uniformly at random. Therefore bagging tends not to work well with linear models [28]. This algorithm as in Algorithm 3.1 is widely used for data sampling with replacement where it generate a number N_b of bootstrapped replicate data $X_t^{b,h}$, where $h = 1,2,\ldots,N_b$of training set X_t that are randomly resampled and replicate a fraction $\theta_b \epsilon [0,1]$ of the total number of training pattern N_t. If the fraction θ_b is large, the training set overlap significantly and the probability for a training pattern to not be in any bagging training pattern is very small. As a result, it might affect the entire training pattern where they are likely to appear in at least one training set and some of them appear multiple times. If the fraction is small, some training set can be absolutely disjointed and some training set pattern might not appear in any training set. Bagging reordering algorithm creates a unique training set with replacement over a uniform probability distribution on the original data. The sampling process with replacement means that each of the sample values are independent where the covariance between two samples is zero.

Algorithm 3.1 *Bagging reordering algorithm .*

```
Input: original dataset DS, number of partition J,
       Boostrap value S
Output: The new training subsets (Tr1, Tr2,...Trm)
Begin
   for t=1 to J
      RandRow =S*rand()
       if RandRow <= S
           St (i, all columns)=
           DS(randRow,AllColumns)
        End if
    Next i
    Output the final training subsets (Tr1, Tr2,...Trm)
End
```

The dynamic reordering alters the pattern sequence at certain times during training in an epoch. Algorithm 3.2 shows the algorithm of dynamic reordering where the training set was randomly drawn without replacement from the original dataset DS associated with the same pattern number, N. The network did not follow the same sequence patterns of epochs because it was randomly reordered at the beginning for each epoch of ANN. This algorithm was adapted from Sospedra [6] where he found a good performance of ensemble networks.

Algorithm 3.2 *Dynamic reordering algorithm.*

```
Input:  original dataset DS, number of partition J
Output: The new training subsets (Tr1, Tr2,...Trm)
Begin
  for e=1 to J
    Generate DSe by sampling DS withoutreplacement
       for t=1 to N pattern
          Tri = xi DSe (randRow, AllColumns)
      end for
   end for
   Output the final training subsets (Tr1, Tr2,...Trm)
End
```

The distributed reordering (DRT) in Algorithm 3.3 was proposed according to the dynamic reordering to enhance the alteration of pattern sequence in certain training in an epoch as dynamic reordering but with considering the available classifier nodes. Given p as the classifier from P classifiers and the random weight initialization for each classifier. The training set was randomly drawn without replacement from the original dataset DS associated with the same pattern number N by referring to the probabilities value of each classifier node. The probability P of classifiers is $\frac{p_i}{N}, i \in [1,..,N]$.

Algorithm 3.3 *Distributed reordering algorithm.*

```
Input: Original dataset DS, number of partition J,
      sampling probability P classifiers 1/N for all J.
Output: The new training subsets (Tr1, Tr2,...Trm)
Begin
   for t=1 to J
        Generate DSe by sampling DS without replacement
        Draw Xi from DS using probability P
         for i=1 to N pattern
           Tri = xi(i,all columns)=
            DS(randRow, AllColumns)
            Adjust probabilities P
        end for
   end for
   Output the final training subsets(Tr1, Tr2,...Trm)
End
```

3.3 Neural network classification

Each ensemble classifier with different datasets partitions, $Tr_1, Tr_2, \ldots Tr_n$ were trained with ANN back propagation (BP) algorithm. The use of the hyperbolic tangent transfer in the hidden layer of the ANN can approximate the mapping between the network's input and output. The final output of the feed forward algorithm will be used as the first phase of back propagation neural network algorithm. This algorithm scheme will minimize the error E function and obtain the new weights and threshold. Two phases were repeated until E converged to a possible minimum value. For example, in this case, the stopping criteria will either achieve an error of 0.001 or up to 10000 iterations. The number of outputs y, which was F_3 was used to indicate whether the ensemble classifiers ware sufficiently reliable for integrating the ensemble members.

3.4 Selection of ensemble networks

A decorrelation maximization method was used as in a previous research [16] to select the suitable number of neural network ensemble members. This method was employed to accomplish the diversity principle in ensemble neural network because the correlation between available neural networks was small. Supposedly, there were p ANN classifiers result $(f_1, f_2, \dots, f_p)$ with n values of output. The error matrix $(e_1, e_2, \dots e_p)$ of p ANN can be represented as:

Matrix 3.1. *Matrix of E error*

$$E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1p} \\ e_{21} & e_{22} & \cdots & e_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ e_{n1} & e_{n2} & \cdots & e_{np} \end{bmatrix}$$

The steps are given as follows:

Step1: Initialization of error matrix $(e_1, e_2, \dots e_p)$ of p ANN

Step2: Compute mean of each e_{ki} , for each output

Step3: Compute variance for each e_{ki} and the desired output e_k

Step4: Compute covariance for each e_{ki} based on step 3 to normalize the output.

Step5: Compute a maximization correlation coefficient p_i , based on (2)

$$p_i^2 = r_i^T R_{-i}^T r_i \ (i = 1,2, \dots, p) \tag{2}$$

where for a pre-specified threshold θ, if $\rho_i^2 < \theta$, then the classifier f_i should be taken out from the p classifiers. On the contrary, the classifier f_i should be retained. This phase involved the multithreading classifier which locates the aggregated output. The weights from each selected multiple ANN networks will be calculated in order to find the average of the connecting weights for ANN. Let D_i be the selected ANN classifier weights, then the weight mean of all selected multiple ANN is:

$$D = \frac{w_1 D_1 + w_2 D_2 + \dots + w_k D_k}{w_1 + w_2 + \dots + w_k} = \frac{\left(\sum_{i=1}^{k} w_i D_i\right)}{\left(\sum_{i=1}^{k} w_i\right)} \tag{3}$$

where k is the size of the dataset, w is the connecting weight of selected multiple ANN. This means that the weight will use a validating vector for the aggregation of outputs generated by multiple ANN.

3.5 Output Aggregation

The aggregation of multiple ANN was applied to combine the output of the ensemble in the form of an aggregated output. The aggregation techniques used to test the cluster of classifier were based on the technique used in a previous study

[29]. These aggregators were selected to be testified due to their strength in ensemble strategies in aggregating the output in most previous research. The first aggregator was the output average where this technique is an average based scheme of an aggregator. It was a simple way to combine the output of the network. The multiple ANN were correlated with each other, and the expected error will be reduced as the number of ensemble members increased after integrating the output. Therefore, the final output was aggregated for this simple average.

Weighted average is the integration of average and weighted majority voting in which the weights are applied not to class labels, but to continuous outputs. This kind of average base combination rule can qualify either as a trainable or non-trainable combination rule, depends on how the weights are obtained [24]. If the weight is obtained during the ensemble generation as a part of the regular training, as in AdaBoost, then it is non-trainable combination rule. If a separate training is used to obtain the weights such as in mixture of experts' model, then the weights are a trainable combination rule. The weights generated for each classifier or for class and classifier the, estimated accuracies from training performances are represented by T weight, $, \ldots, wT$,[30].

Majority voting is the simplest method of voting scheme for combining classifiers. The outputs of certain numbers of individual classifiers are pooled together. The class that receives the largest number of votes is selected as the final classification decision [14]. Majority voting can be in any class whether all classifiers agree, at least one more than half of the classifiers agree, or the highest vote [10]. The disadvantage of this kind of voting is that the information provided by the network is reduced to a single vote so the probabilistic information related to each output is omitted.

Bayesian is among the competitive method used to select the best network for each case of classifiers based on continuous output. The Bayesian aggregator assumes the independence between classifiers, and estimates the class-conditional support to the observed classifier output [31]. The original reference of the Bayesian inference process which represented hypotheses can explain an event E by $H_1, \ldots H_2, \ldots, H_i$. The posterior probability of hypotheses H_i being true is given by the evidence E, $P(H_i | E)$. The decision is based on the values of the posterior probability of the individual classifiers and the network with maximum posterior probability is assigned at a weight of 1 while other network assigned weights of 0. This is important to give a variety process operating regions. The posterior probability of an ANN is as [32],

$$p_t^k = \frac{\frac{1}{\sqrt{2\pi\sigma}} p_{t-1}^k e^{-(\frac{y_t - y_t^k}{\sigma})^2}}{\sum_{m=1}^{n} \frac{1}{\sqrt{2\pi\sigma}} p_{t-1}^m e^{-(\frac{y_t - y_t^k}{\sigma})^2}} \tag{4}$$

σ is the variance of the errors. Equation (4) shows the posterior probability of the current data (at time t) is related to the past posterior probability value (time $t-1$).

The Bayesian inference was used to select the final value of prediction output as that of network with the network with the maximum value of posterior probability at any given time.

4. Result and Discussion

A selection of aggregator has been tested due to their strength in ensemble strategies in aggregating the output for most previous research [30]. MNIST benchmark datasets for handwriting characters has been used as large data representative to test the proposed approach. The datasets are chosen in order to represent the various size of an ANN. MNIST dataset has been divided into three set (50000 for training set, 10000 for validation set, and 10000 for test validation). The techniques were tested using multiple computers which represent the cluster that runs synchronously. All ANN classifiers have been trained with different dataset partitions and parameters. For example, two ANNs work with the same hidden nodes setup, but with different learning rate and momentum, two classifiers using different hidden node setups and learning rate/momentum with resampling techniques for input and etc. The accuracy of the recognition rate obtained from the result was measured by the total corrected recognition.

The Improved Performance (IP) is a measure to find the differences between an incremental ANN and multiple ANN which used ensemble network. This measurement is important in order to obtain the percentage difference of correctly classified patterns in the tested data between a single network and ensemble networks. The calculation was based on:

Definition 4.1 *Improved of Performance (IP).*

$$IP = Performance_{ensemble} - Performance_{singleNet}$$

where the performance is referred to corrected classified patterns of the tested set for multiple ANN and a single ANN. Another measurement was the percentage of reduced error in which shows the differences between the error classified patterns of the tested set for an incremental ANN and the ensemble network.

Definition 4.2 *Percentage of Reduced Error (PER).*

$$PER = 100 \cdot \frac{Error_{singleNet} - Error_{ensemble}}{Error\_singleNet}$$

where the error refers to total classification errors by both methods. Any negative values for IP and PER indicate that multiple ANN for the particular technique performs better than a single ANN.

Fig. 2 shows the mean percentage of IP for four types of reordering ANN with four types of network combiner. For original reordering, majority voting technique recorded the highest IP that was nearly up to 9% for seven classifiers compared to a single classifier. Meanwhile average output recorded 8.6%, and both Bayesian and weighted average scores 5.31% and 3.14% of improvement respectively. For bagging reordering, output average technique recorded the highest increment of Performance which was nearly up to 10% for seven classifiers compared to a single classifier. Both majority voting and Bayesian method scores more than 8% of improved performance. Meanwhile, weighting average method in this case increased 4% of the performance.

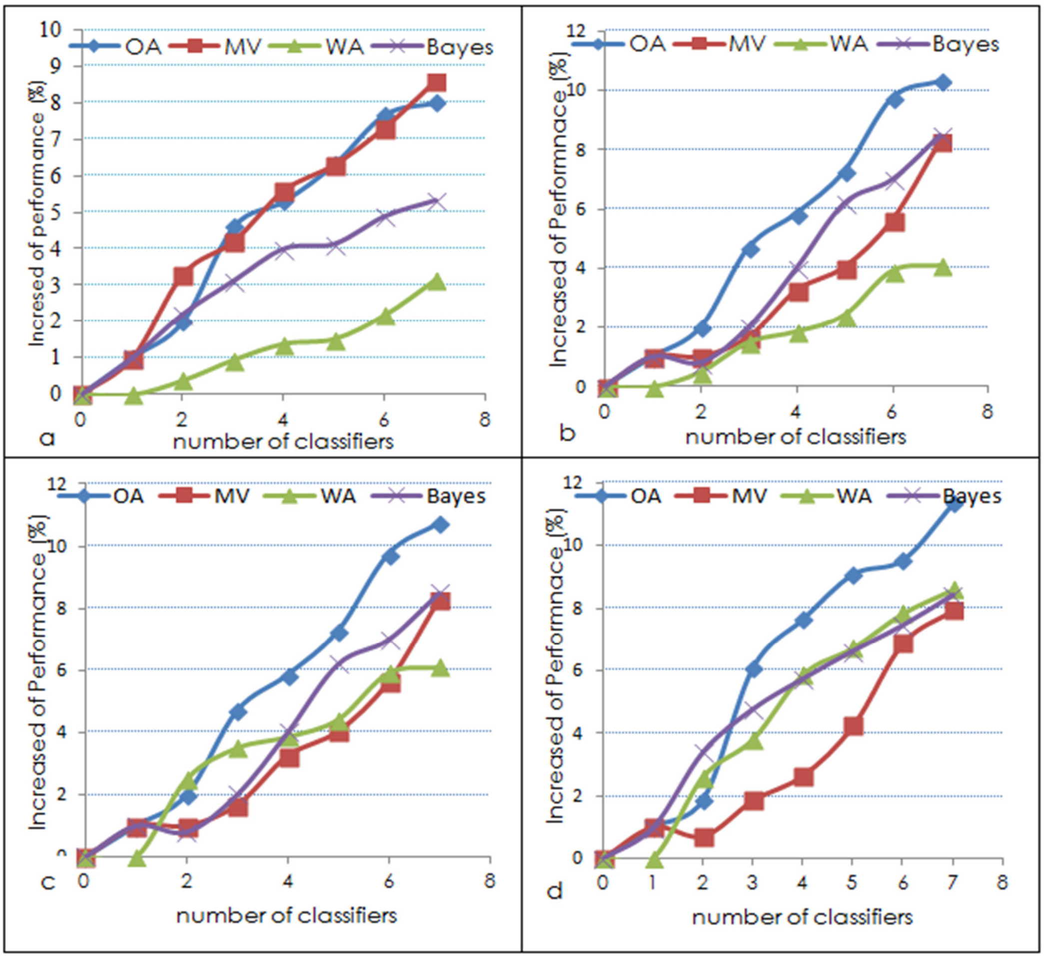

Fig. 2: Increased of Performance for a) Original reordering b) Bagging reordering c) Dynamic reordering and d) Distributed reordering

For dynamic reordering, output average technique recorded the highest increment of performance that was nearly up to 11% for seven classifiers compared to a single classifier. The other methods score more than 8% (majority voting and Bayesian technique) and 6% (weighted average) of improved performance. The improvement of the performance for this network was likely similar to the bagging reordering network where the output average outperformed the other method in which the value was nearly up to 11%. Meanwhile, for distributed reordering, output average technique recorded the highest IP that was nearly more than 11% for seven classifiers compared to a single classifier. Other methods score more than 8% (weighted average and Bayesian technique) and 7% (majority voting) of improved performance.

Fig. 3 shows the percentage of reduced error for original reordering of ANN. The results recorded that the output average technique increased nearly to 53% for this kind of ordering. Meanwhile, the majority voting method scores 40% of error reduction and both Bayesian and weighted average method scores 25% and 19% respectively. The reduced error for this network was not more than 53% but showed differences between each ensemble methods especially when the classifier was increased.

For bagging reordering, the mean percentage of reduced error for the output average technique increased nearly to 69%. This technique has reduced the error by 50% from the use of single network of bagging reordering. Meanwhile, the Bayesian method scores 53%, the majority vote technique scores more than 48%, and the weighted average technique scores 41%. The reduced error scores for majority voting Bayesian method and weighted average have been slightly similar for two to three classifiers. However, it showed significant differences when the classifier increased in the bagging reordering network.

For dynamic reordering, the mean percentage of reduced error for the output average technique for dynamic reordering network increased nearly up to 72%. This technique reduced the error in 58% from the use of single network of distributed reordering. The Bayesian technique scores nearly up to 53%. While both weighted average and majority voting scores nearly up to 50% and 48% respectively. The overall score performance recorded for this network is better than previous two networks. This might be resulted from the non-replacement sampling and partitioning which gave better diversity for each classifier.

The method of output average has recorded the highest performance of increase performance and error reduction for most techniques of multiple ANN with the majority of the ensemble techniques were tested in this experiment. Although, there are limited classifiers in this experiment, it is believed that the network will be doing well for additional classifiers. This result is similar to the results in previous study

[21] in terms of error reduction. There is a reduction of error when the number of classifiers increases. Although it was also shown that the average output technique demonstrated better performance, both majority vote and Bayesian technique are capable of providing good results for any size of multiple ANN with good accuracy scores. Besides showing the average error reduction across the classifiers, the figure also indicates a large span of classification accuracies among the techniques tested.

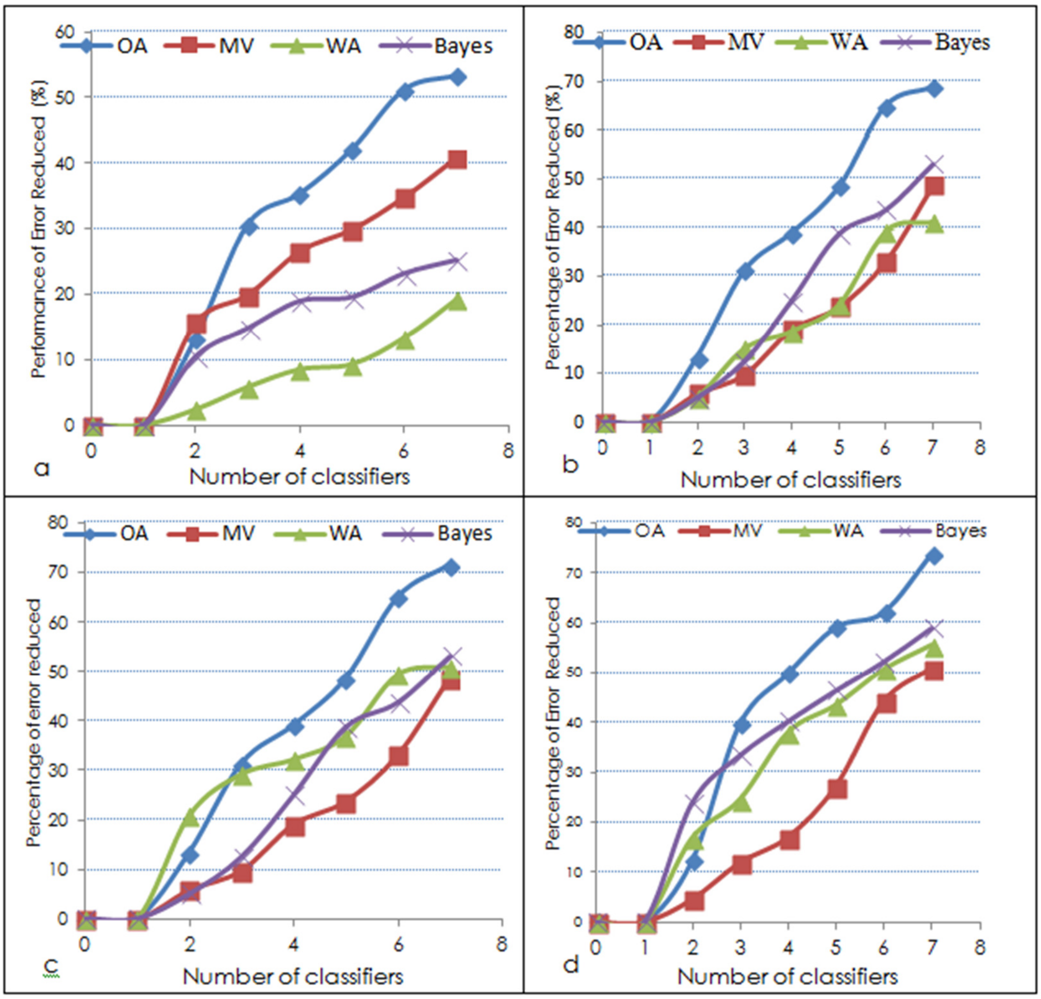

Fig. 3: Performance of Reduced Error for a) Original reordering b) Bagging reordering c) Dynamic reordering and d) Distributed reordering

Distributed reordering ANN showed that the mean percentage of reduced error for output average technique for distributed reordering network had increased more than 73%. This technique has reduced the error by 60% from the use of single network of distributed reordering. The Bayesian technique scores nearly up to 59%, while the scores for both weighted average and majority voting were 55% and 50% respectively. The overall highest performance score for this network was recorded by bagging reordering.

In validating the results, the comparison of each accuracy level was analysed and presented in Table 1. The table shows the comparison percentages of accuracy for each reordering strategies with all combination methods (sample with two, four, and seven ensemble networks). The best score of accuracy percentage was 98.2% by the output average method of distributed reordering ensemble for seven ensemble networks. Ensemble networks that use bagging and dynamic reordering recorded the second and third highest percentage with an output average combiner. Bayesian combiner recorded a good score (97%) for seven ensemble networks with distributed reordering. The distributed reordering technique competes in parallel with dynamic reordering technique in all method of ensemble except for Bayesian combiner. With these many ensemble networks, there is an indicator that uses multiple networks for all ensemble improved.

Table 1: Comparison of ensemble reordering techniques

Ensemble Methods	Ensemble networks number	Ensemble Reordering			
		Original (%)	Bagging (%)	Dynamic (%)	Distributed (%)
Output Average	2	87.6	86.9	86.9	87.1
	4	93.7	92.6	92.9	95
	7	**96.2**	**97.5**	**97**	**98.2**
Mean Diff.		3.7	5.2	4.7	4.8
Weighted Average	2	79.7	83.9	83.9	87
	4	85.4	87.1	87.1	89.7
	7	88.4	92.2	92.2	95.1
Mean Diff.		3.9	4.5	4.5	4.5
Majority Vote	2	86.2	91.3	89.3	87
	4	91.6	93.2	93.2	93.1
	7	94.8	95.5	95.5	95.9
Mean Diff.		3.9	2.2	2.8	3.9
Bayesian Combiner	2	90.7	89.7	89.7	88.3
	4	94.2	94.3	94.3	94.2
	7	90.7	89.7	89.7	**97**
Mean Diff.		-1.2	-1.5	-1.5	3.8

The use of different ensemble networks number also affect the accuracy, where more networks produced better accuracy. The mean difference values showed that the use of output average and weighted average method in all reordering technique in ensemble networks recorded some improvement. Both recorded that the mean difference achieved a value between 3.7 and 4.5 when the networks are bigger. Meanwhile, the majority vote method scored in average for all reordering technique while Bayesian method scored poor value for all reordering technique. Bayesian only scored better when it used distributed reordering with the mean values of difference networks in which scored 3.8 and its highest performance with seven networks was 97%.

Original ordering of ANN provides a minimum ensemble improvement with each method performs similarly to each other in which the improvement percentage is below 10%. Bagging reordering network showed a consistent improvement in performance and reduction errors. Dynamic reordering showed better results than simple reordering in most cases. However, distributed reordering networks presented good performance in terms of reduction error in which the performance improvement was better than dynamic reordering network. All methods showed a variety of performance due to the application of ensemble methods. In this case, average output can be considered the best ensemble method especially for distributed reordering network. In general, the results provided by the ensemble methods with a variety of network pattern ordering diversity were quite similar but there were some specific cases where a combiner performed better on a few sets. This finding was also similar to what [24] have found, in which there is no ensemble method that outperforms other combiner techniques consistently. In addition, the results obtained shared a similar view on the average output technique comparison with what has been demonstrated before [6].

As a summary, for this performance measurement, average output provides a good solution for any ensemble size and method but the other ensemble methods can also provide excellent results for specific network diversity or other datasets. The best performed strategy for sequential large datasets classifier is by using the ensemble network of distributed reordering with Bayesian combiner. Meanwhile the proposed strategy for parallel ensemble classifier for large datasets is by using decorrelated output average method with distributed reordering strategies. The classifiers can also be performed using dynamic reordering for similar combiner (decorrelated output average). The proposed distributed reordering technique also showed high performance for decorrelation of Bayesian technique.

5 Conclusion

Ensemble networks make sense only if the classifiers are diverse. The reordering techniques that promote diversity have contributed to the reduction error and training time for clusters of ANN. The proposed technique had enhanced the generalization ability among different ANN classifiers that provide a promising solution to other binary class classification and recognition problems. The classifier cluster which provides heterogeneous ANN showed good scalability and performance for an ANN multithreading training execution in a case of large dataset. The objective of this paper was achieved where the results showed a satisfactory balance between accuracy and balance in a large dataset classification. It is important to conclude that average output is the first alternative that should be seriously considered in case of the requirement of accuracy and resources is critical. It provides good results for any dataset when applying in ANN ensemble aggregator. In addition, the weighted average procedure is the simplest model. However, further study should take into account all the possible ensemble methods on a particular dataset.

ACKNOWLEDGEMENTS
We thank you to Center of Research and Innovation Management (CRIM), UniSZA, Terengganu, Malaysia.

References

[1] Bottou, L. and Cun, Y. L. 2004. Large Scale Online Learning, Advances in neural information processing systems, Vol.16, 217-224.

[2] Cervantes, J., Li, X., Yu, W., and Li, K. 2008. Support Vector Machine Classification for Large Data Sets via Minimum Enclosing Ball Clustering, Neurocomputing, Vol.71, 611-619.

[3] Tulunay, Y., Tulunay, E., and Senalp, E. T. 2004. The Neural Network Technique- General Exposition, Advances in Space Research, 983-987.

[4] Anthony, M. and Bartlett, P. L. 2009.Neural network learning: Theoretical foundations: Cambridge University Press.

[5] Windeatt, T. 2008. Ensemble MLP Classifier Design, in Computational Intelligence Paradigms. vol. 137, L. Jain, M. Sato-Ilic, M. Virvou, G. Tsihrintzis, V. Balas, and C. Abeynayake, Eds., ed: Springer Berlin Heidelberg, pp. 133-147.

[6] Torres-Sospedra, J., Hernández-Espinosa, C., and Fernández-Redondo, M. 2011. Introducing Reordering Algorithms to Classic Well-Known Ensembles to Improve Their Performance, in Neural Information Processing. vol. 7063, B.-L. Lu, L. Zhang, and J. Kwok, Eds., ed: Springer Berlin Heidelberg, pp. 572-579.

[7] Javadi, M., Ebrahimpour, R., Sajedin, A., Faridi, S., and Zakernejad, S. 2011. Improving ECG Classification Accuracy using an Ensemble of Neural Network Modules, Plos One, Vol.6,

[8] Salkhordeh H., M., Vahedian, A., and Sadoghi Y., H. 2012. Making Diversity Enhancement Based on Multiple Classifier System by Weight Tuning, Neural Processing Letters, Vol.35, 61-80.

[9] Ceamanosa, X., Waske, B., Benediktsson, J. A., Chanussot, J., Fauvele, M., and Sveinsson, J. R. 2010. A Classifier Ensemble Based on Fusion of Support Vector Machines for Classifying Hyperspectral Data, International Journal of Image and Data Fusion, Vol.1, 293–307.

[10]Polikar, R. 2012. Ensemble Learning, in Ensemble Machine Learning: Methods and Applications, C. Zhang and Y. Ma, Eds., ed: Springer US, pp. 1-34.

[11]Ulaş, A., Semerci, M., Yıldız, O. T., and Alpaydın, E. 2009. Incremental construction of classifier and discriminant ensembles, Information Sciences, Vol.179, 1298-1318.

[12]Feng, L., Yao, Y., Jin, B., and Chen, Q. 2012. An improved incremental learning model for network data stream classification problems, Journal of Convergence Information Technology, Vol.7,

[13]Kotsiantis, S. 2011. An incremental ensemble of classifiers, Artificial Intelligence Review, Vol.36, 249-266.

[14]Jain, L., Sato-Ilic, M., Virvou, M., Tsihrintzis, G., Balas, V., Abeynayake, C., and Windeatt, T. 2008. Ensemble MLP Classifier Design, in Computational Intelligence Paradigms. vol. 137, T. Windeatt, Ed., ed: Springer Berlin Heidelberg, pp. 133-147.

[15]Hanafizadeh, P., Parvin, E. S., Asadolahi, P., and Gholami, N. 2008. Ensemble Strategies to Build Neural Network to Facilitate Decision Making, Journal of Industrial Engineering International, Vol.4, 32-38.

[16]Yu, L., Wang, S., and Lai, K. K. 2008. Credit risk assessment with a multistage neural network ensemble learning approach, Expert Systems with Applications, Vol.34, 1434-1444.

[17]Huang, F., Xie, G., and Xiao, R. 2009. Research on Ensemble Learning, International Conference on Artificial Intelligence and Computational Intelligence, No. 249-252.

[18]Wang, S. and Yao, X. 2013. Relationships between diversity of classification ensembles and single-class performance measures, IEEE Transactions on Knowledge and Data Engineering, Vol.25, 206 - 219

[19]Fernández, C., Valle, C., Saravia, F., and Allende, H. 2012. Behavior Analysis of Neural Network Ensemble Algorithm on A Virtual Machine Cluster, Neural Computing & Applications, Vol.21, 535-542.

[20]Minaei-Bidgoli, B., Parvin, H., Alinejad-Rokny, H., Alizadeh, H., and Punch,W. F. 2014. Effects of Resampling Method and Adaptation on Clustering Ensemble Efficacy, Artificial Intelligence Review, Vol.41, 1-22.

[21]Kuncheva, L. and Rodríguez, J. 2014. A Weighted Voting Framework for Classifiers Ensembles, Knowledge and Information Systems, Vol.38, 259-275.

[22]Ciresan, D. C., Meier, U., and Schmidhuber, J. 2012. Multi-column Deep Neural Networks for Image Classification, IEEE Conference on Computer Vision and Pattern Recognition (CVPR) No. 3642-3649.

[23]Alizadeh, H., Behrouz, M.-B., and Parvin, H. 2014. To Improve the Quality of Cluster Ensembles by Selecting a Subset of Base Clusters, Journal of Experimental & Theoretical Artificial Intelligence, Vol.26, 127-150.

[24]Zhou, Z.-H. and Li, N. 2010. Multi-information Ensemble Diversity, in Multiple Classifier Systems. vol. 5997, N. E. Gayar, J. Kittler, and F. Roli, Eds., ed Cairo: Springer Berlin Heidelberg, pp. 134-144.

[25]Woźniak, M., Graña, M., and Corchado, E. 2014. A survey of multiple classifier systems as hybrid systems, Information Fusion, Vol.16, 3-17.

[26]Yang, L. 2011. Classifiers selection for ensemble learning based on accuracy and diversity, Procedia Engineering, Vol.15, 4266-4270.

[27]Yang, Y., Xu, Y., and Zhu, Q.-x. 2010. A neural network ensemble method with new definition of diversity based on output error curve. in Proceeding of Intelligent Computing and Integrated Systems (ICISS), 2010 International Conference on,587-590.

[28]Breiman, L. 1996. Bagging predictors, Machine Learning, Vol.24, 123-140.

[29]Mohamad, M., Saman, M. Y. M., and Hitam, M. S. 2014. The Use of Output Combiners in Enhancing the Performance of Large Data for ANNs, IAENG International Journal of Computer Science, Vol.41, 38-47.

[30]Li, G., Shi, J., and Zhou, J. 2011. Bayesian adaptive combination of short-term wind speed forecasts from neural network models, Renewable Energy, Vol.36, 352-359.

[31]Re, M. and Valentini, G. 2010. Simple ensemble methods are competitive with state-of-the-art data integration methods for gene function prediction, Journal Machine Learning Reserach, Vol.8, 98-111.

[32}Ahmad, Z. and Zhang, J. 2009. Selective combination of multiple neural networks for improving model prediction in nonlinear systems modelling through forward selection and backward elimination, Neurocomputing, Vol.72, 1198-1204.