Data Balanced Bagging Ensemble of Convolutional-LSTM Neural Networks for Time Series Data Classification with an Imbalanced Dataset

Matthew Ward
Center for Southeastern Tropical Advanced Remote Sensing
University of Miami
Miami, FL, USA
mcw140@miami.edu

Kevin Malmsten
Department of Electrical and Computer Engineering
University of St. Thomas
St. Paul, MN, USA
malm4036@stthomas.edu

Hassan Salamy
Department of Electrical and Computer Engineering
University of St. Thomas
St. Paul, MN, USA
hsalamy@stthomas.edu

Cheol-Hong Min
Department of Electrical and Computer Engineering
University of St. Thomas
St. Paul, MN, USA
cmin@stthomas.edu

***Abstract*—A system was developed using a bagging (bootstrap-aggregating) ensemble of neural networks to classify time-series data with class imbalanced datasets. The proposed system uses a Data Balanced Bagging Ensemble (DBBE) of Convolutional-LSTM (CLSTM) Neural Networks (DBBE-CLSTM) to classify accelerometer and EEG datasets. The base neural network (CLSTM) that is used in the DBBE-CLSTM contains three convolutional layers with a batch normalization layer following each convolutional layer, one max-pooling layer, one LSTM layer, and one sigmoid classification layer. A bagging ensemble was created which used the CLSTM paired with the proposed data balancing technique as the ensemble's base learner. The proposed bagging ensemble achieves a validation set average accuracy of 90.42% and a validation set recall of 93.23% on an accelerometer dataset. Ultimately, the proposed DBBE-CLSTM achieves the best overall performance of the models developed in this paper when evaluating both accuracy and recall, with the DBBE-CLSTM achieving greater than 90% for both metrics. As a secondary verification of the proposed DBBE-CLSTM, we evaluated the model on the UCI Epileptic Seizure Detection dataset and show that the model achieved high performance across all evaluated metrics. The DBBE-CLSTM achieved an average validation set accuracy of 99.23%, and average validation set f1-score of 0.9809 on UCI Seizure dataset.**

Keywords—convolutional neural network; time-series data recognition; classification; bagging-ensemble

I. INTRODUCTION

Impacting roughly 1% of individuals worldwide, epilepsy is the most common-place chronic brain disorder affecting the individual's neural connections [1]. In severe cases the abnormal connections in the neural activity may lead to irregular seizures involving uncontrollable convulsions, loss of consciousness, and even death [1]. Due to the risk of these events, monitoring and precautions must be taken for the individual's health and safety. From 2010 to 2015, the US experienced an increase of nearly 700,000 cases of individuals with active epilepsy [2]. Even with medication, these unpredictable interruptions in the daily lives of these people impact their ability to safely live an independent life [1]. For many disorders like epilepsy and Autism Spectrum Disorder (ASD), living an independent life may become difficult.

This study is focused on monitoring and actively identifying binary classified events found in time series data using an ensemble of deep neural networks. The ensemble of deep neural networks is first evaluated on an accelerometer time series dataset collected from individuals with cognitive disabilities. The ensemble of deep neural networks is then verified on the UCI (Bonn) Epileptic Seizure Detection dataset [3, 4]. Section II lays out an introduction to the accelerometer time-series dataset and the UCI dataset analyzed in the paper. Section III describes the proposed DBBE-CLSTM algorithm. Section IV reports the classification metrics and results of the algorithms described in this paper on both datasets. Lastly, Section V contains the Conclusion and Future Work.

II. TIME-SERIES DATA DESCRIPTION

To test the DBBE-CLSTM, two different time series data sets were preprocessed in order to improve the model's accuracy. The first dataset was collected from an accelerometer-based wrist-worn sensor to monitor behavioral patterns of ASD patients while the second dataset is from UCI's epilepsy data set from EEG sensors. The accelerometer time series dataset consists of 811 pairs of time series and their respective binary label (0=no activity present, 1=activity present). Each of the time series is 60 samples long, which corresponds to approximately a 1 second long time series. The UCI dataset consists of 11,500 pairs of time series and their corresponding binary label (0= no seizure detected, 1= seizure detected). Each of the UCI time series is 178 samples long, corresponding to approximately a 1 second long time series.

III. NEURAL NETWORK STRUCTURE

The proposed Data Balanced Bagging Ensemble of Convolutional-Long-Short-Term-Memory Neural Networks (DBBE-CLSTM) is implemented using the Tensorflow [5] and Keras [6] libraries. The below subsections A-C discuss the design of the CLSTM, and then the data balancing technique.

A. Convolutional-LSTM (CLSTM)

The first three blocks of layers in the CLSTM as shown in Fig. 1, consist of convolutional layers, each using the ReLU activation function, followed by a batch normalization layer. Following the first convolutional layer is a max-pooling layer. Following the final batch normalization layer is a LSTM layer. A sigmoid based classification layer completes the model.

2021 IEEE International Symposium on Circuits and Systems (ISCAS) | 978-1-7281-9201-7/20/$31.00 ©2021 IEEE | DOI: 10.1109/ISCAS51556.2021.9401389

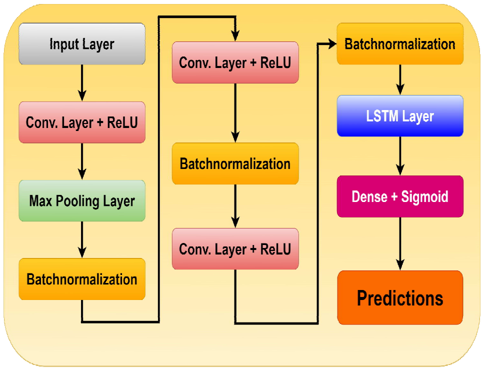

Figure 1. Proposed structure of CLSTM Network

Previous work has shown that a CNN-LSTM neural network can be highly effective for time series classification tasks, and can often outperform a standard CNN [7]. CNN-LSTM can be a highly effective structure for time series classification tasks because of the nature of its components. Convolutional layers can be highly efficient at identifying short to medium length patterns in time series, and LSTM layers can be highly efficient at detecting long patterns. Thus, it can be inferred that CNN-LSTM can achieve high accuracy in time series classification tasks because of its potential ability to detect patterns in time series of varying sizes.

B. *Data Balancing*

Training a neural network with an imbalanced dataset can have a negative impact on the training process and the final performance of the neural network once trained [8]. Firstly, training a neural network with an imbalanced dataset can affect the ability of the neural network to converge to a good solution using the training data. And secondly, training a neural network with an imbalanced dataset can impact the ability of the neural network to generalize to previously unseen datasets such as validation sets or test sets.

In order to improve the stability of the neural network training process, a data balancing technique is used. A binary classification task has two classes: class 0, and class 1. If class 0 has fewer samples than class 1, then in order to train the neural network with a balanced dataset we form a new training dataset by appending all N samples of class 0 and N random samples from class 1. This forms a new balanced training dataset with N samples from each class which can then be used to train the neural network. This can potentially allow for better training of the neural network by improving the optimization process and the neural network's ability to generalize, however, it does discard some of the potentially valuable training data that has already been collected.

C. *Bagging Ensemble*

Because Data Balancing requires the discarding of part of the otherwise usable training data, it can affect the overall validation set accuracy of a neural network that is trained using data balancing. In order to counteract such behaviors, a bootstrap-aggregating ensemble (bagging ensemble) of data balanced Convolutional Long Short Term Memory neural network (DBBE-CLSTM) is developed. Because each individual CLSTM in the ensemble is trained on a random sample with replacement of the training dataset, this ensemble technique can help counteract the decrease in overall validation accuracy that the data balancing technique can produce.

Bootstrap-aggregating, which is utilized in this paper, is a sampling process that involves sampling the training dataset with replacement [9]. The final model developed in this paper is an ensemble of 15 neural networks each trained on a bootstrap-aggregated data balanced training set. The final predictions on the validation set are determined by finding the statistical mode of the predictions of each of the 15 neural networks in the ensemble on the validation set.

IV. System Verification and Results

The primary objective of this work is to develop an algorithm using deep neural networks that can achieve both high overall accuracy and high recall on time series classification tasks with an imbalanced dataset. A secondary objective of this work was to develop an algorithm that is highly parallelizable and computationally efficient to facilitate future work that would involve deploying the algorithm on a group of distributed devices. To achieve the primary objective, an individual CLSTM neural network is developed. In order to further improve the performance of the proposed time series classification algorithm and achieve the secondary objective of this work in creating a highly parallel learning algorithm, a data balanced bagging ensemble learning algorithm (DBBE-CLSTM) is created using the CLSTM developed in this work as its base neural network. After evaluating the performance of both the individual CLSTM neural network as well as the DBBE-CLSTM on Behavior-Time-Series (BTS) Data collected from Autism Spectrum Disorder (ASD) patients the performance of both the individual CLSTM neural network and the DBBE-CLSTM is evaluated on the UCI Epileptic Seizure Detection Dataset and the results are compared to previously published results. All results reported in Tables I-V are the mean results attained from 5-fold cross-validation.

To build a computationally efficient and high performing neural network, our model initially used a small single-layer convolutional neural network and incrementally improved the model to the CLSTM neural network. In order to evaluate the performance of the machine learning models developed in subsections A-F below, we use four metrics: training set accuracy, training set recall, validation set accuracy, and validation set recall to compare our research results with the previously published results.

A. *Standard Scaling*

The first set of experiments were performed using a single layer convolutional neural network with 32 convolutional filters, convolutional window size of 3, and a ReLU activation function. The first analysis evaluated the performance of this single-layer neural network on the BTS data with and without using standard scaling as a data preprocessing technique. From Table I, it can be seen that by including the standard scaling as a preprocessing technique improved the neural network performance considerably for each of the four metrics.

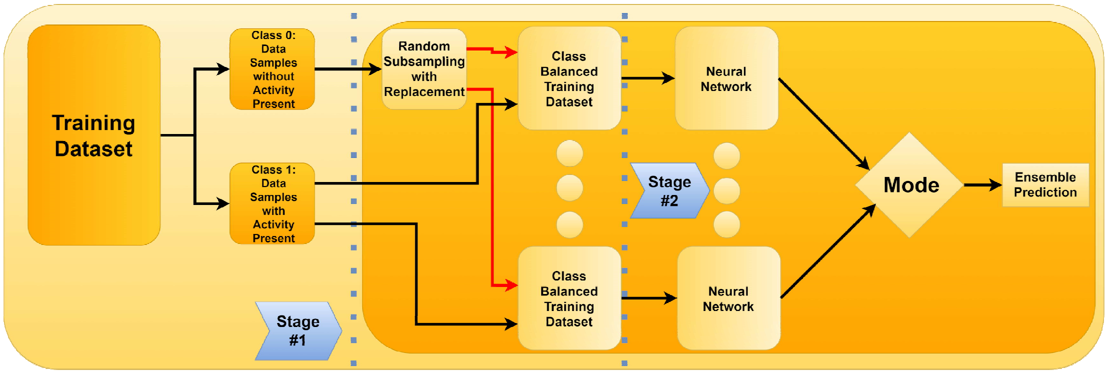

Figure 2. Proposed structure of DBBE-CLSTM Network

B. Convolution Layers and Batch Normalization

Following the experiments shown in Table I, the second set of experiments evaluated the performance impact of increasing the number of convolutional layers in the neural network. This was chosen as the second set of experiments because deeper neural networks have a stronger ability to extract meaningful patterns from data and can, in some instances, achieve higher classification accuracy than a single layer neural network.

Therefore, the experiments shown in Table II evaluate the performance of the neural network when using one, two, three, and four convolutional layers. It can be seen that for both of the validation set metrics being evaluated, the neural network trained with three convolutional layers achieved the best performance. This is likely a direct result of the deeper convolutional neural network's ability to learn more complex patterns in time series. However, it is noted that the neural network that was trained with four convolutional layers performed worse in both validation set metrics than the neural network with three convolutional layers. This likely occurs because the neural network with four convolutional layers is experiencing a significantly worse vanishing gradient problem.

After the experiments shown in Table II were performed, it was recognized that although increasing the number of convolutional layers up to three convolutional layers did increase the classification accuracy, the neural network is still overfitting the training dataset, and increasing the number of layers any further decreases the validation set performance of the neural network. In order to attempt to reduce the impact of the vanishing gradient problem on the neural network and reduce the amount the neural network was overfitting the training data without significantly increasing the complexity of the neural network, the next experiments shown in Table III evaluated the performance of a neural network with three, and four convolutional layers with a batch normalization layer following each convolutional layer. From Table III it can be seen that although for the neural network trained with three convolutional layers the addition of a batch normalization layer did widen the difference between the training accuracy and the validation accuracy, it also boosted the overall validation accuracy attained by 0.1353% and the overall validation recall by 6.34% compared to the results shown in Table II. Additionally from Table III, it can be seen that the neural network trained with four convolutional layers with batch normalization did outperform the neural network trained with four convolutional layers w/o the batch normalization in Table II. However, the neural network with four convolutional layers in Table III does not significantly outperform the one with three convolutional layers and has a larger computational cost. Thus, the neural network with three convolutional layers is selected as the best model from Table III.

TABLE I. DATA PROCESSING TECHNIQUE RESULTS USING STANDARD SCALING

Standard Scaling	Train Accuracy	Train Recall	Valid. Accuracy	Valid. Recall
NO	88.3795%	0.6347	84.4005%	0.5258
YES	**93.1875%**	**0.6944**	**87.1150%**	**0.5419**

C. LSTM

Following the experiments on the number of convolution – batch normalization blocks to use in the proposed neural network as shown in Tables II and III, the next model development stage improved and evaluated the performance of the neural network with a Long-Short-Term-Memory (LSTM) layer added following the third convolution - batch normalization block. This experiment was done because CNN-LSTMs have been shown to outperform standard CNNs as was discussed in section III subsection A. The inclusion of the LSTM layer resulted in a training accuracy of 97.75%, a training recall of 0.9444, a validation accuracy of 93.28%, and a validation recall of 0.8194. From these results, it can be seen that the addition of an LSTM layer increased the validation accuracy by 2.92% and the validation recall by 8.09% compared to the results shown in Table III.

D. Data Balancing

Following the CLSTM experiment shown in subsection C, the next improvement added involved using a data balancing technique with the CLSTM in order to ensure that the neural network was trained with an equal number of training samples

TABLE II. EFFECT OF INCREASING THE NUMBER OF THE CONVOLUTIONAL LAYERS

Layer Count	Train Accuracy	Train Recall	Valid. Accuracy	Valid. Recall
1	93.3725%	0.7177	87.9145%	0.5871
2	96.27%	0.8347	89.0245%	0.6323
3	97.904%	0.9113	**90.5070%**	**0.7129**
4	**98.3355%**	**0.9298**	90.1965%	0.6936

TABLE III. EFFECT OF INCLUDING A BATCH NORMALIZATION LAYER AFTER EACH CONVOLUTION LAYER

Layer Count	Train Acc	Train Recall	Valid Acc	Valid Recall
3	99.1675%	0.9669	**90.6295%**	0.7581
4	**99.3530%**	**0.9782**	90.5045%	**0.7677**

that belonged to activity class 0 and 1. The result of this experiment is as follows: a training accuracy of 94.9195%, a training recall of 0.9686, a validation accuracy of 87.363%, and a validation recall of 0.9387. From these results, it can be seen that the data balancing technique, which is a form of undersampling, reduced the overall validation set accuracy attained. However, the data balancing technique did increase the validation set recall obtained by the CLSTM by 14.56% compared to the results shown in subsection C.

E. Max Pooling

Following the experiments that evaluated the CLSTM performance with data balancing, a single max-pooling layer was included in the CLSTM after the first convolutional layer and the impact using the following four metrics was recorded: classification accuracy, recall, training time, and inference time. The inclusion of a single max-pooling layer in the CLSTM resulted in a training accuracy of 96.3708%, a training recall of 0.9689, a validation set accuracy of 89.3938%, and a validation set recall of 0.9161. Including one pooling layer in the neural network reduces the training time by 4.1934 seconds and the inference time by 0.0173 seconds. From these results, it can be seen that the inclusion of a max-pooling layer increased the validation set accuracy by 2.32% compared to the results shown in subsection D and reduces both the training and inference time. However, including one pooling layer reduced the validation set recall by 2.41%. The reported times were recorded using the Tensorflow 2.3.0 library running on a machine equipped with a 2.9 GHz Dual Core Intel Core i5 CPU, and 8 GB of RAM.

F. Data Balanced Bagging Ensemble

Following the experiments shown in subsection E, the final set of experiments performed in an attempt to increase both the validation accuracy and the validation recall was to evaluate the number of neural networks (each neural network has the structure shown in Fig. 1 and the performance reported in subsection E) to use in a data balanced bagging ensemble (shown in Fig. 2). Table IV shows that the bagging ensemble, which contained 15 neural networks achieved the highest overall performance, with a validation accuracy of 90.421% and a validation recall of 0.9323. This is a 1.15% increase in validation accuracy and a 1.77% increase in validation recall compared to the best results in subsection E. The algorithms shown in Table IV are the only algorithms described throughout this work which achieve both a validation accuracy as well as a validation recall greater than 90%, making these the best and final models developed in this work.

TABLE IV. EFFECT OF DATA BALANCED BAGGING ENSEMBLE

Number of Neural Networks	Pooling Layers	Train Acc	Train Recall	Valid Acc	Valid Recall
5	1	92.3087%	0.9761	89.8652%	0.9312
10	1	**92.9303%**	0.9758	**90.4210%**	0.9258
15	1	92.7605%	**0.9766**	**90.4210%**	**0.9323**

TABLE V. EVALUATING THE PERFORMANCE OF THE DBBE- CLSTM ON THE UCI EPILEPTIC SEIZURE DATASET

#NN	1NN	10NN	10d-NN
Train Accuracy	99.2445%	99.2845%	**99.5363%**
Train Recall	0.9919	0.9941	**0.9960**
Train Precision	**0.9931**	0.9709	0.9811
Train F1	**0.9924**	0.9823	0.9885
Val Accuracy	98.3418%	98.9096%	**99.2286%**
Val Recall	0.9814	0.9848	**0.9885**
Val Precision	0.9397	0.9616	**0.9734**
Val F1	0.9597	0.9731	**0.9809**

G. UCI Epileptic Seizure Detection Dataset

As a final verification of the performance of the DBBE-CLSTM, it is evaluated on the binary-case UCI Epileptic Seizure Detection dataset. For these experiments, class 1 (epileptic seizure) remained labeled as class 1, and classes 2,3,4,5 (no epileptic seizure) are all labeled as class 0 in our binary classification problem. In Table V the column labeled as "1NN" evaluated the performance of a single data balanced CLSTM. The column labeled "10NN" evaluates the DBBE-CLSTM with 10 data balanced CLSTM in the ensemble. Lastly, the column "10d-NN" evaluates the DBBE-CLSTM with 10 data balanced CLSTM in the ensemble when a detrending procedure is applied to the UCI dataset before being input to the neural network. The DBBE-CLSTM 10d-NN model achieves 2.09% higher validation set accuracy and 4.77% higher validation set recall than a recently proposed stacking ensemble of deep neural networks does on this dataset [10].

V. CONCLUSION AND FUTURE WORK

Upon our initial evaluation of the proposed DBBE-CLSTM algorithm on both the accelerometer time series dataset and the UCI Epileptic Seizure Detection dataset, we have shown that the DBBE-CLSTM algorithm is a high performance algorithm for binary class imbalanced time series classification tasks. In addition, we can also conclude that the proposed DBBE-CLSTM is a highly parallelizable learning algorithm because each neural network in the DBBE-CLSTM is fully independent of one another, i.e. there is no communication between them. Therefore, the DBBE-CLSTM would be well suited for distributed computing.

We plan to further improve the DBBE-CLSTM algorithm developed in this work by further exploring different activation functions, enhancing the optimization algorithms, and exploring different ensembling method to be used. In addition, the DBBE-CLSTM algorithm will be evaluated on additional datasets including the multiclass UCI Epileptic Seizure Detection dataset. Finally, we plan to evaluate training the DBBE-CLSTM algorithm on a distributed network of devices.

REFERENCES

[1] A. Krukowski, I. Lamprinos, B. C, J. Rodriguez and E. Vogiatzaki, Cyberphysical Systems for Epilepsy and Related Brain Disorders, 1st ed. Springer International Publishing, pp. 11-38, 2015.

[2] Center of Disease Control and Prevention, "Active Epilepsy and Seizure Control in Adults - United States, 2013 and 2015," 2018.

[3] RG. Andrzejak, K. Lehnertz, C. Rieke, F. Mormann, P. David, CE Elger, "Indications of nonlinear deterministic and finite dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state," Phys. Rev. E, 64, 061907, 2001.

[4] EEG Dataset Repository http://epileptologie-bonn.de/cms/front_content.php?idcat=193&lang=3&changelang=3

[5] Tensorflow, https://www.tensorflow.org/

[6] Keras, https://keras.io

[7] G. Swapna, Kp Soman, and R, Vinayakumar, 2018. "Automated detection of diabetes using CNN and CNN-LSTM network and heart rate signals," Procedia Computer Science, 132, pp. 1253-1262, 2018.

[8] M. Buda, A. Maki, and M. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," Neural Networks, 106, pp. 249-259, 2018.

[9] C. Aggarwal, Neural Networks and Deep Learning. Springer International Publishing, pp. 186-187, 2018.

[10] K. Akyol, "Stacking ensemble based deep neural networks modeling for effective epileptic seizure detection," Expert Systems with Applications, vol. 148, p. 113239, 2020. Available: 10.1016/j.eswa.2020.113239.