Bi-Directional Neural Networks and Genetic Algorithms for Classifying Physiological Signals

Tom Willis,

Research School of Computer Science, Australian National University,
Canberra, Australia
u6377372@anu.edu.au

Abstract. Neural Networks can be a useful tool for decision making, although trust and reliability are called into question when we rely on technology to inform experts. This study compares the performance of a typical Neural Network to a Bi-direction Neural Network, trained and evaluated on a dataset consisting of pupil dilation features of an observer. The neural networks classify the severity of depression of an individual who is being watched by the observer. We aim to show that a Bi-direction Neural Network is suitable for this task and hence could bring its benefits to professional fields where decision tools need to be reliable. Previous work on this dataset yielded an average accuracy of 92%. This paper's baseline gave an accuracy of 35% while the Bi-directional Neural Network performed at 40% accuracy.

Keywords: Depression Detection, Physiological Signals, Pupil Dilation, Bi-directional Neural Network, Artificial Intelligence Trust, Genetic Algorithms.

1 Introduction

A recent paper has demonstrated a method of diagnosing depression with reduced bias, since the method does not rely on the patient describing their symptoms or answering questions with sincerity and honesty [1]. The researchers' method involved capturing physiological signals expressed by observers who were watching videos of a person talking [1]. These videos showed someone who by current depression diagnostic methods expressed from none or minimal depression, to severe depression [1]. The researchers were able to train a classifier Neural Network on features extracted from these physiological signals with 92% accuracy [1].

Through this method, the researchers implemented a method for diagnosing depression in a way that minimises bias since the diagnosis is derived from unconscious physiological signals. The method also places less demand on the potentially unstable patient to reflect on their experiences, instead requiring them to complete simple tasks such as reading to a camera. This method has benefits that should be further explored.

Doctors, patients and other stakeholders require a level of trust and scrutiny in the tools that are used to diagnose illnesses, however there is currently a lack of trust in artificial intelligence systems, presenting a barrier to overcome before artificial intelligence could be adopted in healthcare [2]. A way to rectify this is to extract meaning from the Neural Network models, which is seen as a way to improve the acceptability of Neural Network [3]. In this field, extracting meaning would also give the Neural Network the ability to explain its diagnosis so that a medical professional can give concrete reasons for their subsequent diagnosis.

A previously suggested Neural Network architecture that has the potential to extract meaning from Neural Network models is called a Bi-directional Neural Network (BDNN). In a paper, researchers demonstrated this Model architecture on two datasets, demonstrating its ability to analyse the data [4]. A Bi-directional Neural Network is a network that is trained on data features as input and on target class values as input, meaning the network is sometimes trained in the reverse direction [4]. This imitates the brain's electrical synaptic transmission, where the transmission is usually bi-directional, enabling the Neural Network to remember input patterns and output vectors [4,5,6]. By remembering both input patterns and output vectors the Neural Network can start building relationships between the two, which consequently was demonstrated by the researchers could be used to extract meaning [4], and hence has potential to provide medical professionals with information that supports diagnosis.

Our goal is to explore the use of a BDNN on the dataset that was used to classify patients severity of depression, to do the same task. Through this, we will investigate how the use of a BDNN affects model performance and hence whether such an architecture and method combination could be used to help diagnose depression and provide doctors with the model's reasoning. We also use Genetic Algorithms (GAs) for feature and hyperparameter selection.

This paper examines the data being used for classification, devises the structure and procedures of Neural Networks trained for this classification task, and compares model performance between models. We conclude the paper with a discussion of results and suggesting future work to further explore the problem.

2 Method

2.1 The Data

The data was collected through sensors attached to participants who watched 16 videos, each video of which depicted an individual reading a paragraph or answering a set of questions [4]. The individual depicted fit into one of four depression categories and of the 16 videos, there is an even spread of individuals in all four categories of depression [4]. Of the dataset, there are 12 participants who each watched all 16 videos producing 192 patterns of data [4].

Each participant was attached to 3 different sensors to measure 3 different physiological traits as the participants watched the videos [4]. Features of the dataset are derived and categorized by these 3 sensors. The sensors measured:

- Galvanic Skin Response, a measure of electricity flow through the skin which is affected by the amount of sweat on the skin [4].
- Skin Temperature, indicative of blood flow to the participants peripherals [4].
- Pupil Dilation, a measure of pupil size which may indicate a participants response to emotional engaging stimuli [4].

A large part of the paper which created this dataset focused on feature selection, with the goal of finding which features most contributed to a NN's decision. The researchers found that a NN trained on all the pupil dilation features was marginally worse then their best model that was trained on a subset of features from all sensors [4], indicating that most of the features that contributed to their NN's decision were pupil dilation features. Hence, this study will only consider pupil dilation features. Reducing the amount of features we use by choosing to use features that gave the best results, removes a lot of the features that didn't affect the results. By removing these features, we avoid giving the nn information that is irrelevant, leading to a smaller NN that takes less time to train.

To further describe the data set, the first column is the participant identifier. This was used to split data based on the participant and not used as a training feature. The second column is the observed persons depression diagnosis class, and hence is also the target output class. The remaining columns are all pupil dilation features that serve as pattern inputs to the Neural Networks. These features were preprocessed from signal data by the original researchers. These features include the minimum, maximum, mean, standard deviation, variance, root mean square, means of the absolute values of the first and second difference metrics of normalised pupillary size and average pupillary size of the left and right eyes [1]. A very low pass and a low pass filter were constructed and applied independently to the pupillary dilation signal of both eyes by the original researchers [1]. Features extracted from these signals include the number and average amplitudes of peaks and ratio of peaks between the very low pass and low pass filters [1].

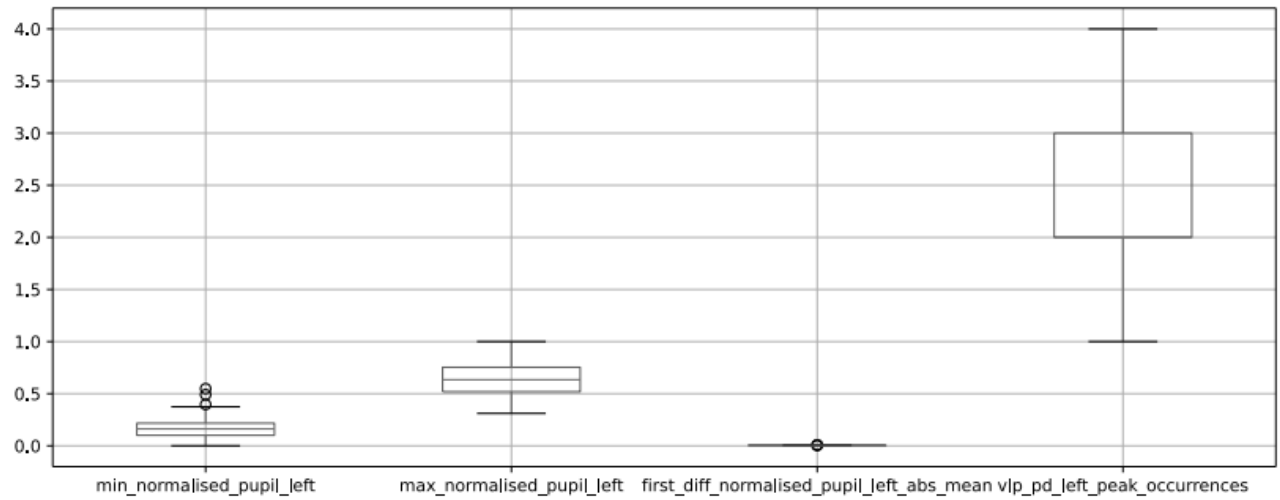

Fig. 1. A box and whisker representation of a subset of features in the dataset

The value ranges of these features differ. Some are integer numbers and some are decimals close to zero. These differing ranges could cause a neural network to learn the features at different rates. To address this, this paper further processed the dataset through normalisation using min-max scaling.

2.2 The Baseline Method

To evaluate how well a BDNN classifier performs against a typical NN classifier on the dataset, we must first devise this typical NN. To make this as comparable to the original depression experiment as possible, I built the NN as close to this classifier as possible. The NN devised is fully connected with 1 hidden layer and 4 output neurons representing the 4 output classes. The model uses a sigmoid activation function which performs equally or better than the relu activation function. This NN and all other models were trained using the Adam optimizer using back propagation. This model used the Cross-Entropy loss function.

The study split the available training data into two, 80% of the rows for training the model and 20% of the rows for evaluating the model. The rows are split up randomly. We split the data since the model will likely make a better prediction on data it's been trained on than unseen data. Making predictions on seen data would bias the evaluation since it doesn't demonstrate the model's ability to generalise and focus on the important features. A model that generalises well will make good predictions on unseen data.

Evaluating a model involves comparing what the model predicted compared to the ground truth. Due to random initial weight array initialisation model performance can vary. So when evaluating hyper parameters, three models are created and evaluated. This study calculated model accuracy, precision, recall and F1 score for each output class and then averaged all these metrics across evaluations to get the average model accuracy, average precision, average recall, and average F1 score. These are the same metrics calculated by the researchers who created the depression dataset. This study also calculates the standard deviation of model prediction accuracy.

After hyper parameter tuning has completed, ten models are created and evaluated on unseen data. In order to provide this unseen data this study implemented Leave-One-Participant-Out. Leave-One-Participant-Out removes one participant's data from the training dataset and is not trained or validated on. We conduct a forward pass on this data to evaluate the model's performance using the previously outlined metrics.

2.3 The Experimental Method

The experimental model is an implementation of a Bidirectional Neural Network (BDNN). This type of network can make predictions in two directions. In the forward direction we give the model a row of data and the model will predict it's class. In the reverseForward direction we give the model a class and the model returns a row of data that fits this class. The BDNN completes a reverseForward pass by executing the layers and subsequently its weight matrices in reverse order. To train a model like this we need to complete training in both directions.

To train the network completes a combination of forward and reverseForward passes. It then calculates the network's loss and applies the error back-propagation technique to adjust the weight matrix of the network. Deciding when the network completes these passess and how the network calculates loss is considered to be part of hyper parameter tuning.

The different methods of training the BDNN that this study explores include:

- Method 1: complete forward and reverseForward passes on the same epoch.
- Method 2: switch the training direction after a specified number of epochs.

Losses are calculated using Mean Squared Error. Through method 1, the loss is calculated by adding the loss of both passes together. Through method 2, the loss between forward and reverseForward passes are calculated separately.

2.4 Feature Selection Method

To select features to give the model a genetic algorithm (GA) was used. The GA evaluates a population over generations of individuals that represent the dataset features used to train a model. Each model had input neurons equal to the number of selected features. The features used in the model were represented by a list of single bits. When mutation occurred each bit had a random chance of flipping. Models were given an initial good set of parameters that were discovered through manual experimentation. These hyper parameters were 50 hidden neurons, 3000 training epochs and a learning rate of 2. The GA for feature selection was run on both the NN and the BDNN. A set of features was discovered that performed best on both model architectures. A representation of the optimal set of features can be found in the appendix.

2.5 Hyper Parameter Selection Method

A genetic algorithm was also constructed to explore model hyper parameters and was run on both models after running the feature selection genetic algorithm. This method was also run on the baseline model and given all features. For the

NN model the hyperparameters being explored were the number of hidden neurons, training epochs and the learning rate. For the BDNN, the GA also explored hyper parameters related to the two different methods of training. These hyper parameters are which training method to use, and if the second is chosen, how many epochs before a swap in training direction. These hyper parameters were represented by a range of integers for each hyperparameter. When mutation occurred, each parameter had a chance of changing to another value within the range for that hyperparameter. After finding good hyperparameters, the feature selection GA was run again using these selected hyper parameters however no better feature set was found.

Both genetic algorithms evaluated the models three times to retrieve their average fitness. This fitness is the validation accuracy plus half of the train accuracy. We use both of these accuracy's in the fitness calculation so that the algorithms select models that learnt well on the training data and predict well on unseen data.

Table 1. Genetic Algorithm parameters

GA Parameter	**Value**
Population Size	40
Crossover rate	0.8
Mutation rate	0.4
Crossover type	Two point
Selection type	Tournament of size 10

2.6 Summary

In summary, one participant's data is used for testing, the rest of the data is split into train and validation sets. Model's weights are initialised randomly so many models are trained and their performances averaged in order to evaluate each architecture. Three model architectures are created. The baseline model is a NN trained with the best found hyperparameters. Another NN and a BDNN are trained both on the best found hyperparameters and the best found feature sets for each model. These best found hyper parameters and feature sets are found by two Genetic Algorithms.

5 Results

3.1 Performance on Normalised Data

First results this paper analyses are the results of the NN model when given normalised features. The study ran the GA for finding Hyperparameters on this, however the best average accuracy found was 17.5% and best overall accuracy at 31.5%. This average accuracy is less than chance at 25% and less accurate than observers classification of the people depicted in the videos at 27% [1]. Better accuracy is found with models given the non-preprocessed dataset. Henceforth, models use the non-preprocessed dataset, the same dataset used by the original researchers.

3.2 GA Hyperparameter Results & Discussion

The best found hyperparameters discovered by the genetic algorithms are displayed in table 2.

Table 2. Hyper parameters found by genetic algorithm search

	Baseline NN	NN + GAs	BDNN + GAs
Hidden Neurons	86	86	59
Training Epochs	3582	3582	7914
Learning Rate	2	2	2
BDNN Training Method	-	-	Method 1 (forward and reverse forward pass every epoch)

After the Genetic Algorithm explored the BDNN's hyperparameters, only training method 1 appeared in the hall of fame. This could indicate that the second method didn't make good predictions compared to the first and hence was out performed. To confirm this, the study forced the model to use training method 2 when exploring hyperparameters. This allows the GA to explore the second training method without being dwarfed by the first, and could find good hyper parameters for training method 2. Although no such hyper parameters were found, therefore training method 1 performed objectively better.

The BDNN uses less hidden neurons than the NN. This is likely shows that the two direction training of the BDNN helps the model generalise better and learns less of the noise in the data that can only be found in one direction, leading to less internal features being learnt and hence less neurons are required. The larger amount of training epochs is likely due to training in two directions creates a more broad loss that takes longer to learn the internal features.

3.3 Model Performances

The three classification models were trained and evaluated on the pupil dilation physiological dataset. The accuracy, standard deviation, precision, recall, and F1 score were all calculated and averaged for each model. These results are shown in table 3.

Table 3. Performance of measures for depression classifier models defined from pupil dilation physiological signals.

Depression level	Baseline NN			NN + GAs			BDNN + GAs		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
None	0.28	0.28	0.27	0.59	0.78	0.66	0.57	0.40	0.43
Mild	0.67	0.23	0.32	0.22	0.20	0.33	0.20	0.18	0.19
Moderate	0.37	0.48	0.41	0.65	0.75	0.68	0.32	0.38	0.34
Severe	0.42	0.36	0.35	0.48	0.50	0.47	0.56	0.66	0.60
Average Accuracy	35.625			50.00			40.63		
Standard Deviation	9.34			7.22			7.93		

Of the three models the NN with GA's made the best predictions overall achieving an average accuracy of 50%. This model made the best predictions on the "None" and "Moderate" depression severities and the BDNN with GA's made the best predictions on the "Severe" class. No model was particularly good at classifying the "Mild" class. It is interesting to note that in the original study on this dataset, the participants were also worst at classifying the "Mild" class [1]. In terms of overall accuracy, the BDNN with GA's on average suffered a 9.37% accuracy drop when compared to the NN with GA's. This demonstrates that the BDNN model on average performs 18.74% worse than the NN.

The baseline NN was constructed in aim of replicating the model created by researchers at the Australian National University (ANU). The model, created by researchers at the ANU, was trained on all the pupil dilation features and achieved an average accuracy of 92% [1]. This is the same accuracy that they achieved when combined with a genetic algorithm [1]. In this study, we achieve worse performance across both models that replicate the work by the ANU researchers. In contrast this study created a performance increase when the genetic algorithm is applied.

While in this study we do not replicate the same high accuracy achieved by the ANU researchers, we do demonstrate a potential bound in performance when comparing BDNN and NN models on this dataset. If a NN model was created with high performance on this dataset, when the same techniques are used to create a BDNN, we can expect the BDNN to perform approximately 20% worse. This is a large drop in performance and may demonstrate that a BDNN is not suitable to make predictions on and for explaining its reasoning for predictions in the medical field.

5 Conclusion

This research first created a Neural Network trained on observers pupil dilation data with the goal of predicting the depression severity of an observed individual. The Network performance falls short of the original experiment with this similar goal showing a failure to replicate results. What this study has demonstrated is that it is possible to train a

Bidirectional Neural Network with similar but worse performance than a standard Neural Network on this depression dataset.

6 Future Work

Due to the negative result of this study, future work with this goal should explore one of two avenues. Improving on the methods utilised in this study, with the aim of achieving more accurate models with the BDNN performance more comparable to a typical NN. Or by exploring further methods to make NNs more useful in the medical field such as alternate methods of obtaining learnt neural network reasoning.

References

1. Zhu, X., Gedeon, T., Caldwell, S., & Jones, R. (2019), Detecting emotional reactions to videos of depression. In INES'19: IEEE 23rd International Conference on Intelligent Engineering Systems (6 pp).
2. Asan, O., Bayrak, A. E., & Choudhury, A. (2020), Artificial Intelligence and Human Trust in Healthcare: Focus on Clinicians, Journal of medical Internet research, vol. 22, no. 6.
3. Bochereau, L. and Bourgine, P. (1990), Expert systems made with neural networks, International JointConference on Neural Networks, vol. 2, pp. 579-582.
4. A. F. Nejad and T. D. Gedeon, (1995), Bidirectional neural networks and class prototypes, Proceedings of ICNN'95 - International Conference on Neural Networks, pp. 1322-1327 vol.3.
5. Edelman, G. M., Gall, W. E. and Cowan, W. M. (1987). Synaptic Function, New York: Wiley.
6. Kandel, E. R., Siegelbaum, S. A. and Schwartz, J. H. (1991). Synaptic Transmission, Principles of NeuralSciences.

Appendix

Optimal dataset features found by the genetic algorithm include the following feature names:

min_normalised_pupil_left, max_normalised_pupil_left, mean_normalised_pupil_left, std_normalised_pupil_left, second_diff_normalised_pupil_left_abs_mean, lp_pd_left_peak_occurrences, ratio_peak_occurrence_vlp_lp_left, mean_normalised_pupil_right, std_normalised_pupil_right, rms_normalised_pupil_right, second_diff_normalised_pupil_right_abs_mean, vlp_pd_right_peak_occurrences, lp_pd_right_peak_occurrences, mean_normalised_pupil_avg, var_normalised_pupil_avg, rms_normalised_pupil_avg, first_diff_normalised_pupil_avg_abs_mean, vlp_pd_avg_peak_occurrences, lp_pd_avg_peak_occurrences, ratio_peak_occurrence_vlp_lp_avg.