Detecting Depression Levels using Deep Equilibrium Cascade with Feature Selection using Genetic Algorithm

Chinh Duc La

The Australian National University
Canberra, Australia
u7098799@anu.edu.au

Abstract. Depression is a serious mental illness problem. Detecting it too late could result in someone's life. With the increasing trend in using neural networks for medical diagnosis, we try to use a neural network to detect depression levels using physiological signals from observers who watched some videos of depression patients. We use Genetic Algorithm to select features from dataset. Then, use Cascade Network[13] with implicit layers inspired by Deep Equilibrium Networks [1]. The performance of our model is surprisingly low with average accuracy of 31.7% as we suspect this might be a problem from the dataset.

Keywords: Depression Detection · Neural Network · Implicit Layer · Cascade Network · Genetic Algorithm · Feature Selection.

1 Introduction

Depression is one of many serious mental illnesses in the modern world that could result in someone's death. However, depression is hard to detect as its symptoms are hard to notice, unlike many other conventional diseases. Detecting depression at early stages could save lives and maybe a better treatment result for depression patients. Normally, the diagnosis process requires a lot of questions from experts to detect depression levels. Sometimes, the diagnosis result also depends on how the patient's answers. If we could have a different approach that requires less time and less expertise but reliable, we could have a better treatment for depression patients.

Neural network has been showing that it could perform at the superhuman level in some specific tasks. Many are trying to use neural networks for medical applications. For example, [3] [10] try to diagnosis COVID-19 from chest X-ray images or [9]try to improve the healthcare system by predicting what to do next from Electronic Health Records or [5] could classify skin cancer at the dermatologist level using neural network. With the advancement in physiological technology, [16] could classify depression levels from physiological signals of observers with the accuracy of 88% using only a simple neural network architecture and 92% with the help of Genetic Algorithm to select features. In this paper, we will try to beat the previous result of [16].

Feature selection could be seen as a combinatorial optimization problem with the objective of finding the best combination of input features that maximize some predefined goal. An exhaustive search for a subset of features that satisfy the problem could cost $O(2^N)$ where N is the number of input features. However, we could find a local optimal solution for this problem using Genetic Algorithm, an algorithm that was inspired by natural selection. Genetic Algorithm could find a really high quality solution through biological inspired operations like mutation, selection and crossover.

This paper do the following:

- Using Genetic Algorithm to select features from dataset that work the best.
- Using data and training process from [16] to train and evaluate performance.
- Using the subset of features that is found by Genetic Algorithm to train Cascade neural network [13] with modification from Deep Equilibrium Network [1] with the hope for better performance.

2 Methodology

2.1 Dataset

We will use a subset of data from [16], which consists of 192 data point (unlike the original data has 240 data point) that were extracted from physiological signals of 12 observers (original experiment has 14 observers) who watched videos of different depression level patients and rated depression level from 0 (no depression) to 3 (severe depression). Physiological signals were captured during the experiment are:

– Galvanic Skin Response (GSR): convert into 23 features after feature extraction process.
– Pupillary Dilation (PD): convert into 39 features after feature extraction process.
– Skin Temperature (ST): convert into23 features after feature extraction process.

We have a total of 85 features extracted from all 3 signals. Upon inspecting dataset we have found something interesting. We use K-means Clustering to cluster the dataset and use 2 first principal components to visualize the dataset. From Figure 1 we can see that our clustering result and the label provided by the dataset don't really match. Labels from the dataset likely to be random from the whole data space while the K-means clustering result is almost nicely clustered. The 2 clusters that are overlapping each other could be Mild and Moderate depression levels since [16] states that the performance of their model is not really good at these two categories or we can understand it as data points from these two categories are not nicely separated. As [16] states, "our observers were not very good at consciously identifying the depression level of individuals in videos. . **The overall accuracy was 27%** ...". We think that the **label given in our dataset might be the label that was identified by observers, not the true labels**.

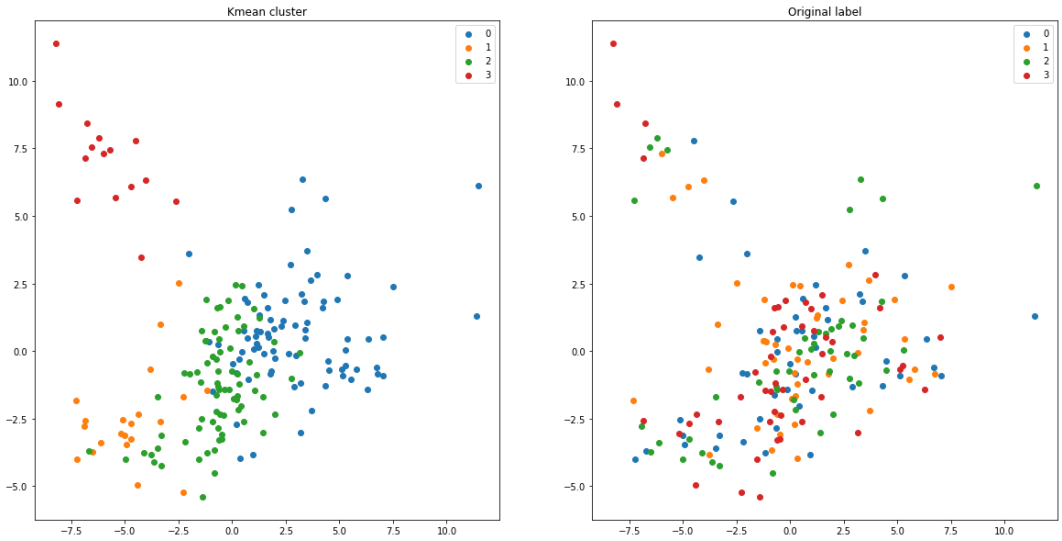

Fig. 1. Left picture show clustering by K-means and original label (right picture) after being projected into 2 dimensions space by PCA

2.2 Casper Algorithm

Casper Algorithm [13] is a constructive algorithm designed to overcome the limitation of Cascade Correlation algorithm (Cascor) [6]. Casper constructs the cascade networks the same way as Cascor: start with one hidden neuron and add a new hidden neuron after the loss function stops decreasing in the training process. But instead of freezing previous neurons that have been trained in trainning process like Cascor, Casper continues to train previously added neurons along with the newly added neuron but with adaptive learning rates based on the position of each neuron. The neural network will be divided into 3 regions:

– **L1**: Connection from input, previously added neuron to the newly added neuron.
– **L2**: Connection from the newly added neuron to output of neural network
– **L3**: All the remaining connections.

The learning rate for each region is set to be $learning\_rate_1 >> learning\_rate_2 > learning\_rate_3$, where $learning\_rate_i$ is the learning rate for region i. The original paper uses resilient backpropagation (RPROP) but in this work, we will use RMSPROP [12] to train our neural network.

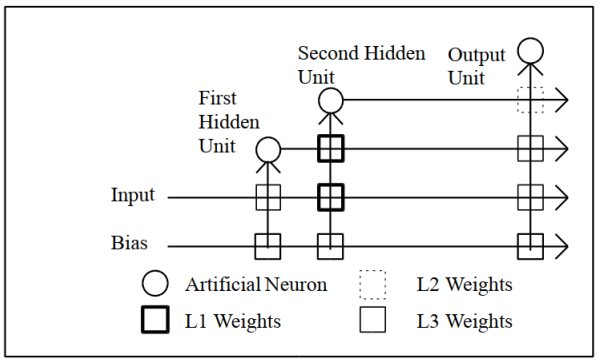

Fig. 2. Casper architecture with 3 seperate regions (image from [6])

2.3 Implicit Layers

Recent years have shown the success of implicit layers, a new class of layer of neural network where instead of computing the output explicitly as a function of input, implicit layer is defined as a problem satisfying some joint condition between input and output. Some of the problems could be differential equations [4], fixed point iteration [1] or optimization solutions [7]. Deep equilibrium models could be understand as we keep stacking layers until the output does not change no matter how many layers we keep adding. Due to cascade topology is somewhat similar to Deep equilibrium models (DEQ) [1]. And [2] had shown that network multiple layers that share the same weight could perform at the same level as different weight, we consider each hidden neuron of the cascade network is a fixed point iteration problem, which we could see as stacking many layers with same weights. In this work, We will use Anderson Acceleration[14] to find fixed point of implicit layer (DEQ layer).

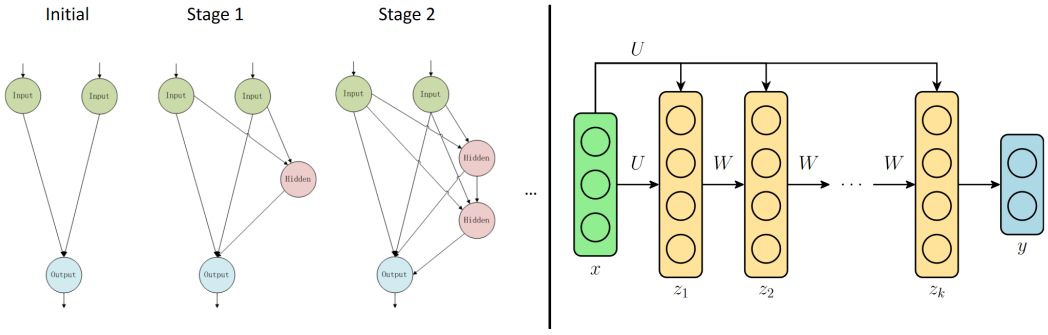

Fig. 3. Cascade topology (left) with 2 neurons share the same topology with DEQ models (right) with 2 layers

2.4 Genetic Algorithm for Feature Selection

Genetic Algorithm (GA) is an algorithm that is inspired by natural selection. Genetic Algorithm is commonly used for optimization and search problems by relying on biologically inspired operators such as mutation, crossover and selection. In this work, we will use Genetic Algorithm to find a subset of features that could increase the model's performance. We will use 1 in the chromosome as the feature is in the subset, and 0 as it's not like showing in figure 4. We use accuracy as a fitness function, and other settings for Genetic Algorithm is specified as Table 1.

2.5 Evaluation

We will use Recall, Precision and F1 score as evaluation metrics for each depression level as our model evaluation. We use *leave-one-patient-out* cross-validation for model selection. As when we train our model, we choose one patient's data for test, one patient's data for validation and the rest for training. Then, choose model with highest validation result to run on our test patient to get report result.

Feature	F1	F2	F3	F4	F5	F6	F7	...	Fn
Chromosome	1	1	1	0	1	0	0	...	1

Fig. 4. An example of representation of chromosome for feature selection with Genetic Algorithm. Feature $F1, F2, F3, F5, Fn$ is in the selected subset, $F4, F6, F7$ is not in the chosen subset.

Table 1. Genetic Algorithm Parameters

Population	100
Generation	80
Fitness	Model's accuracy
Crossover rate	0.8
Mutation rate	1/85
Crossover type	Uniform crossover
Mutation type	Uniform mutation
Selection type	Stochastic universal sampling

3 Experiences

3.1 Implementation details

We normalize data to have a mean of zero and a standard deviation of 1, to have a faster training process [8] [15]. We use leave-one-patient-out cross validation to train neural networks and test their performance on Google Colab. We will implement 3 models:

- **Baseline**: 1 hidden layer neuron network with 50 hidden neurons and sigmoid activation function and train with Adam optimization algorithm like in [16] and Dropout [11].
- **Casper**: a cascade neuron network with 40 hidden neurons and train with RMSProp
- **Fixed Point Casper**: A cascade neuron network with 2 hidden neurons. Each neuron is an implicit layer to solve fixed point problem and train with RMSProp

We use GA to find 12 different subsets for 12 folds of cross validation and combine 12 best bitstrings representing the presence of 85 features by counting how many times each feature appear in the best bitstrings then choose a threshold to choose whether or not to keep the feature (for example, if we choose threshold as 6 which means we only choose features that appears at least 6 times from all 12 best subsets). Finally, we train 3 models again with the choosen subset and evaluate performance.

3.2 Results

Training with full dataset Table 2 show our experiment result. The final result is surprisingly lower than we expect. The baseline was implemented in [16] has an average accuracy of 88% while our best baseline model only has the performance of 32.2%. Recall, Precision, and F1 score are also low. F1 score is low because our model usually decided that it's best to label all input with the same label, it makes Recall or Precision of that performance is zero which makes F1 score is also zero.

Our Casper's performance is even worse, with 23.9% in average accuracy. However, Casper has shown that it could detect better with None and Severe depression levels and really bad at the other two. Fixed Point Casper-perform quite well compared to baseline model in general although it but it is smaller in size.

When we tried to observe output from our models, we also realized that in many cases, both our Casper models also learned that it's best to give a label for all input data. It makes us suspect the dataset might be the problem.

Training with subset feature found by GA Table 3 show the performance of our models when using a subset of feature generated by GA. Overall performance of all three models increases with Casper has a jump of 3.1% in accuracy (from 23.9% to 27%). Baseline and Fixed Point Casper has a slight increase in average accuracy with 1.6% and 0.4% respectively. However, Baseline still have the highest performance comparing to the other two. This could mainly because we choose Baseline's performance as the fitness function for GA. Nonetheless, although we tried to find a subset that reflects the decision of the observers, our models' performance are still just better than randomly guessing by a small margin. Which means the distribution of each features with respect to each depression levels seems to be random. Therefore trying to find an optimal subset is not feasible.

Table 2. Performance measure for depression recognition models defined from all Physiological signals

Depression level	Baseline			Casper			Fixed Point Casper		
	Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score
None	0.249	0.229	0.165	0.099	**0.354**	0.147	**0.293**	**0.354**	**0.226**
Mild	**0.287**	0.270	0.239	0.056	0.083	0.056	0.179	**0.333**	0.209
Moderate	0.218	**0.33**	0.205	0.046	0.125	0.06	**0.313**	0.25	0.189
Severe	**0.424**	**0.458**	**0.332**	0.102	0.395	0.162	0.24	0.313	0.228
Average	**0.294**	**0.322**	**0.235**	0.076	0.239	0.106	0.256	0.313	0.213
Overall Accuracy	**0.322**			0.239			0.313		

Table 3. Performance measure for depression recognition models using a subset of features generated by GA

Depression level	Baseline+GA			Casper+GA			Fixed Point Casper+GA		
	Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score
None	0.284	**0.562**	**0.333**	0.041	0.166	0.066	**0.322**	0.437	0.292
Mild	0.127	0.062	0.055	0.134	0.187	0.111	**0.141**	**0.229**	**0.134**
Moderate	**0.370**	0.291	**0.279**	0.104	**0.416**	0.166	0.188	0.208	0.137
Severe	**0.373**	**0.437**	**0.340**	0.113	0.312	0.157	0.344	0.395	0.280
Average	**0.288**	**0.338**	**0.252**	0.098	0.270	0.125	0.249	0.317	0.211
Overall Accuracy	**0.338**			0.270			0.317		

4 Ablation Studies

Number of hidden neurons and model's performance Although [1] stated that stacking DEQ layers does not increase the performance, the topology of Cascade network is still a little bit different. We tried to increase the total number of neurons of Fixed Point Casper to see if the performance of our model increase. The answer is no. As can be seen from figure 5, when the number of hidden neurons increases, the accuracy of the model fluctuates around 25%-26%.

Finding a subset of features that representing observers' prediction After inspecting the dataset, we have a hypothesis that the label in our dataset is not the true labels from the original dataset but the label the observers predicted. We want to find a subset of features that could reflect the prediction of observers better. However, the result is not as we predicted. The right figure of Figure 6 shows the number of features to keep with respect to each threshold. The left figure of Figure 6 shows something really interesting:

- As the number of feature decreases, the average accuracy of Baseline increases, Fixed Point Casper decreases. The reason might be because we use accuracy of Baseline as the fitness function for GA.
- Casper fluctuates around 25% regardless the number of input features.

Modifying Casper to Fixed Point Casper changes the behavior of original Casper structure. More input features benefit Fixed Point Casper more than less input features. But, less irrelevant input features benefit Baseline. As a result, we could have a new development for this structure which is adding a hidden layer before start adding Fixed Point Cascade. The hidden layer could work as feature extraction from the subset to get more useful information so Fixed Point Cascade could work better.

5 Conclusion and Future Works

We tried to Caper Algorithm and Fixed Point Casper to classify depression levels from physiology signals. Comparing to the Baseline model, Casper takes a longer time to train and Fixed Point Casper takes an even longer time with the same number of hidden neurons. Fixed Point Casper could perform with similar level to baseline model with smaller parameters (only 2 hidden neurons). However, we suspect that provided data might be incorrectly labeled so we cannot evaluate our model's performance. As a result, we tried to find a subset of features from the dataset that we believe that might reflect the way dataset was labeled using Genetic Algorithm. The Performance did increase but not as much as we expect. But, Fixed Point Casper works better when there are more input features while Baseline work better when choosing only relevant subset. So we could add a hidden layer before adding Fixed Point Cascade Neuron to make use of the infromation extraction from a single hidden layer. Another problem we want to address is the label of the provided dataset which we believe is incorrect, we could do some data preprocessing

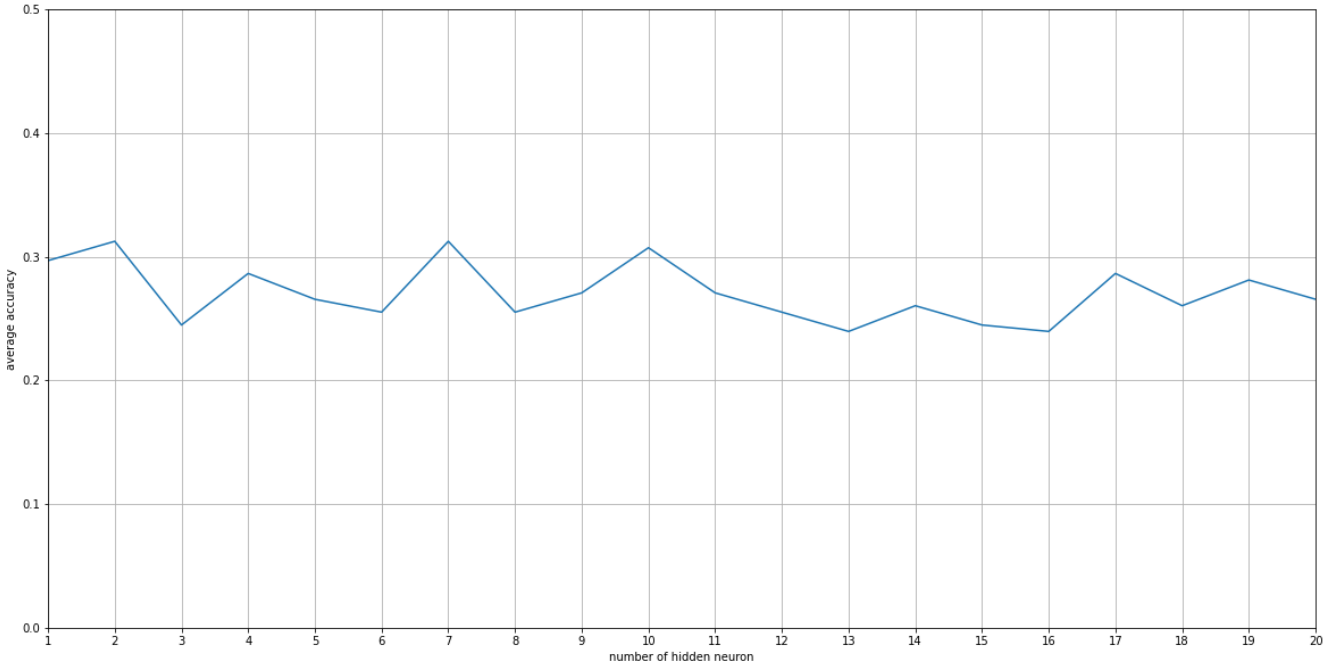

Fig. 5. Number of neurons and average accuracy of Fixed Point Casper

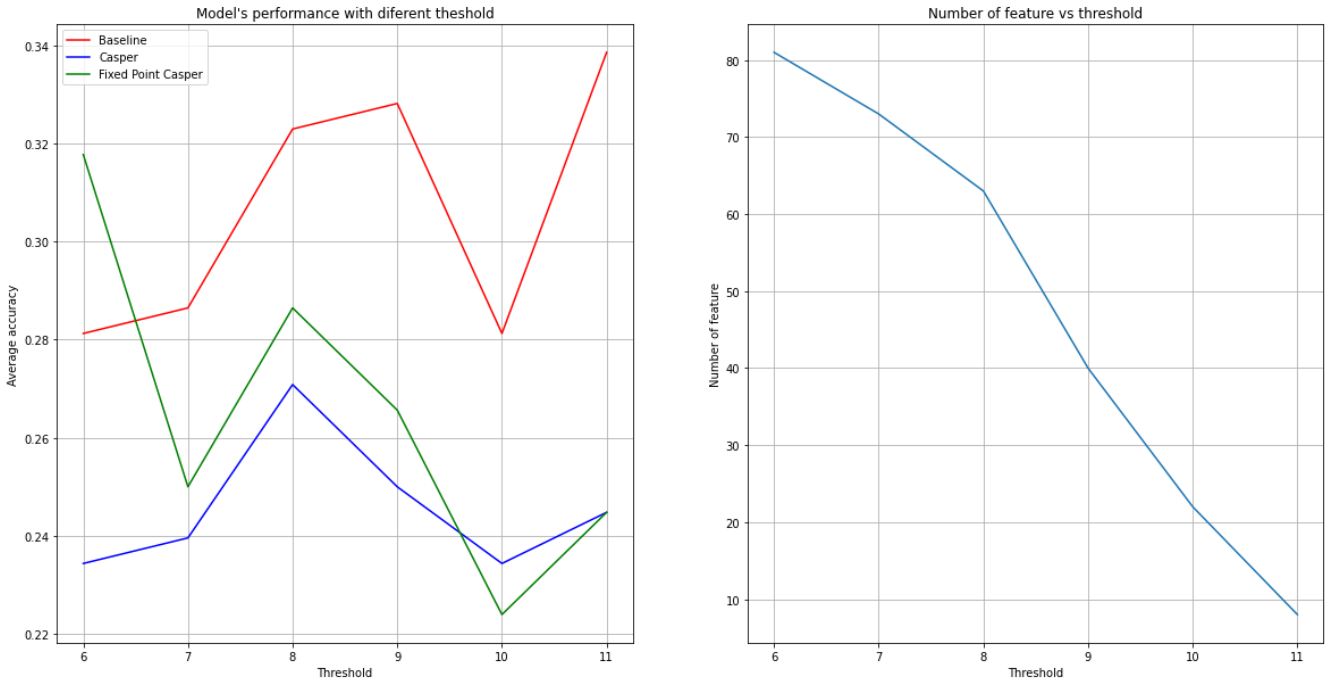

Fig. 6. Caption

using information from [16] such as the highest the accuracy of prediction from observer with respect to each label and Kmeans to try to recover the true label.

References

1. Bai, S., Kolter, J.Z., Koltun, V.: Deep equilibrium models. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
2. Bai, S., Kolter, J.Z., Koltun, V.: Trellis networks for sequence modeling (2019)
3. Bassi, P.R.A.S., Attux, R.: A deep convolutional neural network for covid-19 detection using chest x-rays. Research on Biomedical Engineering (Apr 2021). https://doi.org/10.1007/s42600-021-00132-9, http://dx.doi.org/10.1007/s42600-021-00132-9
4. Chen, R.T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D.: Neural ordinary differential equations (2019)
5. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**(7639), 115–118 (Feb 2017). https://doi.org/10.1038/nature21056, https://doi.org/10.1038/nature21056
6. Fahlman, S.E., Lebiere, C.: The Cascade-Correlation Learning Architecture, p. 524–532. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1990)
7. Gould, S., Hartley, R., Campbell, D.: Deep declarative networks: A new hope (2020)
8. LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.R.: Efficient BackProp, pp. 9–48. Springer Berlin Heidelberg, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35289-8_3, https://doi.org/10.1007/978-3-642-35289-8_3
9. Rajkomar, A., Oren, E., Chen, K., Dai, A.M., Hajaj, N., Hardt, M., Liu, P.J., Liu, X., Marcus, J., Sun, M., Sundberg, P., Yee, H., Zhang, K., Zhang, Y., Flores, G., Duggan, G.E., Irvine, J., Le, Q., Litsch, K., Mossin, A., Tansuwan, J., Wang, D., Wexler, J., Wilson, J., Ludwig, D., Volchenboum, S.L., Chou, K., Pearson, M., Madabushi, S., Shah, N.H., Butte, A.J., Howell, M.D., Cui, C., Corrado, G.S., Dean, J.: Scalable and accurate deep learning with electronic health records. npj Digital Medicine **1**(1), 18 (May 2018). https://doi.org/10.1038/s41746-018-0029-1, https://doi.org/10.1038/s41746-018-0029-1
10. Saha, P., Sadi, M.S., Islam, M.M.: Emcnet: Automated covid-19 diagnosis from x-ray images using convolutional neural network and ensemble of machine learning classifiers. Informatics in Medicine Unlocked **22**, 100505 (2021). https://doi.org/https://doi.org/10.1016/j.imu.2020.100505, https://www.sciencedirect.com/science/article/pii/S2352914820306560
11. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**(56), 1929–1958 (2014), http://jmlr.org/papers/v15/srivastava14a.html
12. Tijmen, T., Geoffrey, H.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude (10 2012), cousera: Neural Networks for Machine Learning
13. Treadgold, N.K., Gedeon, T.D.: A cascade network algorithm employing progressive rprop. In: Mira, J., Moreno-Díaz, R., Cabestany, J. (eds.) Biological and Artificial Computation: From Neuroscience to Technology. pp. 733–742. Springer Berlin Heidelberg, Berlin, Heidelberg (1997)
14. Walker, H.F., Ni, P.: Anderson acceleration for fixed-point iterations. SIAM J. Numer. Anal. **49**(4), 1715–1735 (Aug 2011). https://doi.org/10.1137/10078356X, https://doi.org/10.1137/10078356X
15. Wiesler, S., Ney, H.: A convergence analysis of log-linear training. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems. vol. 24. Curran Associates, Inc. (2011), https://proceedings.neurips.cc/paper/2011/file/e836d813fd184325132fca8edcdfb40e-Paper.pdf
16. Zhu, X., Gedeon, T., Caldwell, S., Jones, R.: Detecting emotional reactions to videos of depression. In: 2019 IEEE 23rd International Conference on Intelligent Engineering Systems (INES). pp. 000147–000152 (2019). https://doi.org/10.1109/INES46365.2019.9109519