Explaining Long Short Term Memory Neural Network Predictions Using Characteristic Inputs and Extracted Rules

Jamie Matthews[1],

[1] Research School of Computer Science, Australian National University"
{Jamie Matthews, u6102798@anu.edu.au}

Abstract. It is difficult to understand the reason that a long short term memory neural network predicts a particular output for a given input. This paper describes the development and testing of a system to provide explanation mechanisms. These explain the reason for the prediction by comparing to the most similar characteristic input and if that fails, a backup ruleset extracted from the net. This method was applied to a neural network which uses a person's eye movement to predict their guess about whether an image is manipulated.

Keywords: Characteristic inputs; Rule extraction; Neural network explanation, LSTM

1 Introduction

The experiments were performed on a recurrent neural network trained on the Caldwell Image Manipulation Eye Gaze dataset, which contains 372 image manipulation guesses. The goal of the neural network was to predict the guess of a participant about whether an image was manipulated. The network was given which of the participants was guessing, the image used, whether the image was manipulated, and gaze fixation data (Caldwell et al. 2015). A sample of the data is shown in Tables 1 and 2. The 'per-guess' data contains a sequence of length 'num-fixs' rows for each row in the 'raw metadata'.

Table 1: Raw metadata

participant	num_fixs	fixs_dur	num_man_fixs	man_fixs_dur	image	image manipulated	vote
1	134	23.212	5	0.85	10	1	1
2	61	18.176	3	2.016	10	1	0
3	108	33.603	16	6.349	10	1	1
4	62	14.137	2	0.218	10	1	0
5	93	28.944	5	1.8	10	1	0
6	142	29.203	12	4.815	10	1	2
7	87	29.691	13	2.149	10	1	0
8	44	29.281	7	5.067	10	1	0
9	162	36.568	6	0.767	10	1	0
11	102	30.533	5	1.699	10	1	0

Table 2: Per-guess data

Participant	Image ID	X Pos	Y Pos	Start Time	Stop Time	Duration	Samples in fixation
1	10	406	764	989.699	989.765	0.066	5
1	10	408	736	968.438	968.505	0.067	5
1	10	427	724	968.555	968.622	0.067	5
1	10	447	423	959.375	959.475	0.1	7
1	10	468	424	943.746	943.863	0.117	8
1	10	472	456	958.175	958.558	0.383	24
1	10	478	440	958.642	958.791	0.149	10
1	10	479	542	959.041	959.125	0.084	6

The purpose of the exercise is to explain why the network predicted a particular guess. This can be difficult when only given the input and output data, and so requires explanation mechanisms. These mechanisms are based on those from T. Gedeon and H. Turner's paper 'Explaining student grades predicted by a neural network' (1993). The explanation procedure methodology was followed except for the method of rule extraction. Instead of a full causal index based on characteristic inputs, we instead use the ruleset extracted from a decision tree train on the predictions of the neural network.

2 Method

A neural network was trained on a pre-processed copy of the data. The data was shuffled to prevent ordering from interfering with results. The 'vote' column of the raw data was used as the label, and the sequence of 'per-guess' data corresponding to the row of each vote in the 'raw metadata' were used as the input features. The data was then copied and one copy was normalised to prevent the network from being biased towards certain features. The non-normalised data was kept for the purpose of being used to explain the network using the original data instead of the normalised data.

The neural network needs to be capable of handling variable-length sequences. Therefore we use a long short-term memory (LSTM) network to process the data.

Hyperparameters. The LSTM used was a custom pytorch LSTM neural network with four hidden layers of 500 neurons each. The hidden layers used the pytorch LSTM function. There were thirteen input neurons corresponding to the thirteen columns in each row of the sequence. The output neurons used a linear activation function. There were three output neurons corresponding to the values of the label column 'vote' – zero: which indicated the participant believed there was no manipulation, one: which indicated the participant believed there was manipulation, and two: which indicated the participant was unsure.

Due to the low number of uncertain votes (16), the network never predicted uncertainty. Therefore explanation mechanisms only used the certain predictions.

The neural network was trained with a learning rate of 0.001 over 1500 epochs. Cross entropy loss was used as the loss function, and pytorch's implementation of Adam was used as the optimiser.

In order to determine these hyperparameters, a combination of ray-tune automatic parameter search and manually applied gradient descent were used. During determination, k-fold cross validation was used in order to prevent training-test splits causing biasing and inaccuracies.

2.1 Explanation Mechanisms

Characteristic Inputs. The primary explanation mechanism is that of the characteristic input. Characteristic inputs are produced per output of the neural network. The set of inputs which cause a particular output is used to create a characteristic input for that output. Many methods can be used to translate the set of inputs into a single characteristic input. In this work, as in the predecessor paper mention above (T. Gedeon, S. Turner. 1993) we use the arithmetic mean of the vector components to create the characteristic inputs.

When explaining why the neural network predicted a particular output for some input, the input is compared to the characteristic inputs. The closest characteristic input by Euclidean distance is selected, and the output is given as the predicted output. That is, it is assumed that the neural network will classify inputs close to a characteristic input as having the same output.

Extracted Rulesets. There may be cases where the characteristic input method produces an output inconsistent with the neural network's actual prediction. In order to explain these cases, we use more detailed rules extracted from the neural network as explanation mechanisms for the prediction. There are many methods to extract rules from neural networks including sensitivity analysis (A. Engelbrecht, H. Victor. 1999) and causal attribution (Chattopadhyay et al. 2019).

Here, we use rules extracted from a decision tree trained on the neural network's predictions. Limiting the decision tree to a lower depth reduces the accuracy with which the ruleset imitates the network but increases the ease with which the ruleset can be understood and followed. A depth of three was found to be an acceptable balance between accuracy and understandability.

2.2 Explanation Procedure

For some input we can explain why the neural network predicts an output.

1. Run the input through the neural network to get the actual output.

2. Liken the input to the characteristic inputs and select the closest.

3. In the case that the characteristic comparison fails, we apply the extracted ruleset to the input.

Step two will give the characteristic input corresponding to an output most likely identical to the actual output. This provides an explanation – the input is most like this characteristic input; therefore, it is predicted to give the same output.

In the case that the output from step two is different from that in step one we perform step three. This will assign an output to the input based on the ruleset extracted from the neural network. The ruleset can be understood and manually applied to explain the reason that a predicted output is chosen.

3 Results

3.1 Explanation from Characteristic Inputs

```
Compare example input to characteristic input
Input to Explain:
         [74,    56,     17.55, 2,     0.433, 13,    0]
Predicted Output: 0
Characteristic inputs:
      0: [45.71, 72.96,  17.02, 8.32,  1.8,   11.95, 0.34]
      1: [19.29, 112.86, 25.61, 43.12, 10.67, 12.02, 0.99]
Distance between example input and characteristic input 0 is 33.63
Distance between example input and characteristic input 1 is 89.94

Example input is closest to [45.71, 72.96, 17.02, 8.32, 1.8, 11.95, 0.34]
which is the characteristic input which gives an output of 0
```

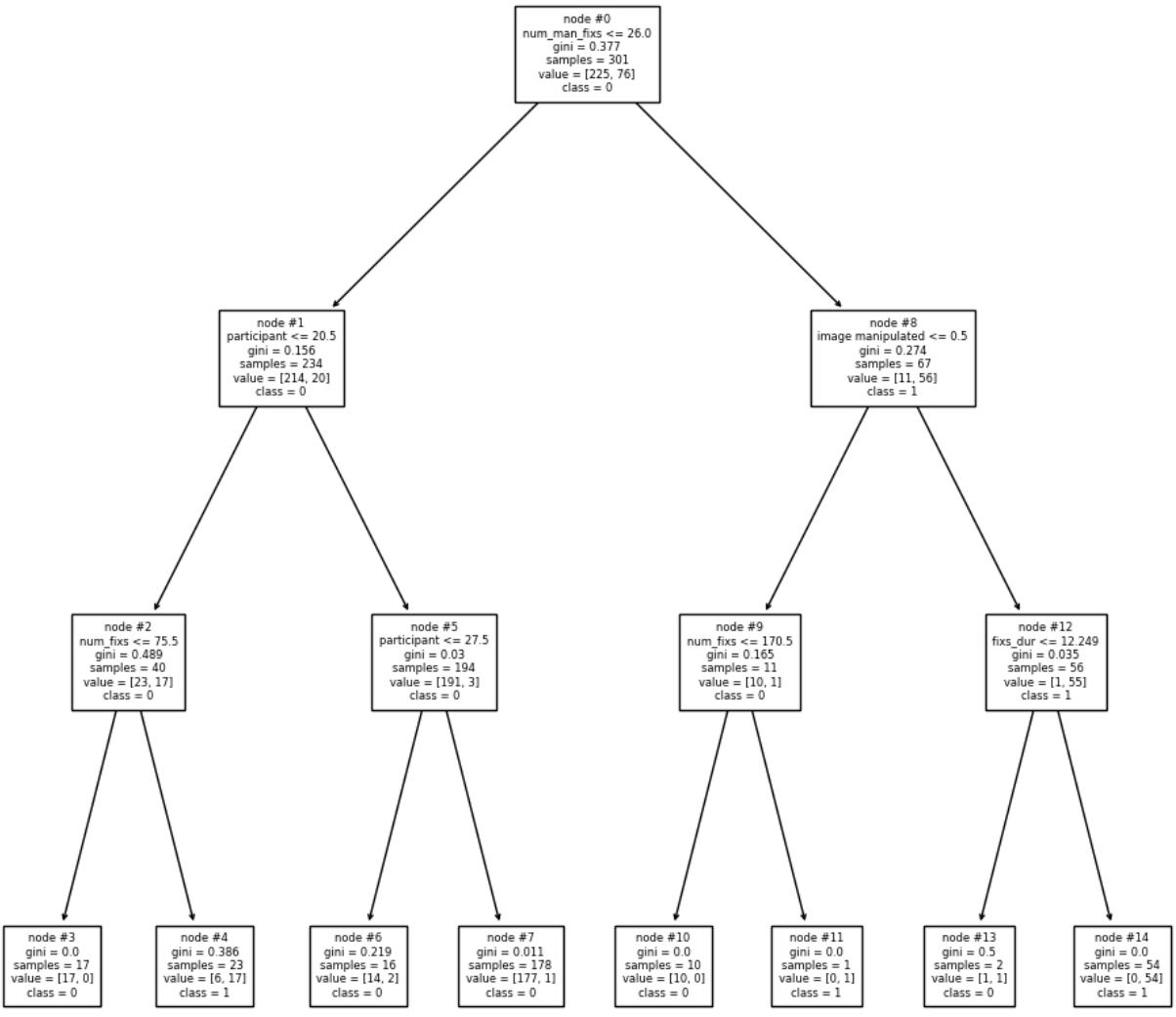
Explanation example 1: Characteristic Inputs

Explanation example 1 shows how comparing inputs to the characteristic inputs can produce easily followed explanations. The example input is obviously closer to the zero characteristic input, and therefore the result logically follows.

3.2 Explanation from Extracted Ruleset

```
Input to Explain:
      [ 67,    130,    37.564, 9,     5.183, 10,    1]
Predicted Output: 0
Characteristic inputs:
   0: [46.02,  70.91,  16.48,  8.22,  1.78,  11.92, 0.32]
   1: [19.91,  114.65, 26.37,  39.59, 9.78,  12.06, 0.98]
Distance between example input and characteristic input 0 is 66.28
Distance between example input and characteristic input 1 is 59.49

Example input is closest to
      [19.91,  114.65,  26.37,  39.59,  9.78,  12.06,  0.98] which is the
characteristic input which gives an output of 1
This is different from the neural network's actual prediction. We will use
rules extracted from the neural network to get a better answer
The input lands in node number 7 - this is an output of 0.
```



Explanation example 2: Extracted Ruleset

Explanation example 2 demonstrates a failure of the characteristic input comparison, and the resulting fallback to applying the extracted ruleset. The ruleset application can be manually followed and understood.

4 Conclusion

A neural network was created in order to predict the votes made by participants in the study conducted by Caldwell et al. (2015). To explain why the neural network made any given prediction we use explanation methods. These methods enable us to understand the relationship between the input features and the predictions. The methods require extraction of characteristic inputs and datasets.

4.1 Future work

More advanced methods of extracting characteristic inputs – such as weighting the influence an input has over the characteristic input based on the level of activation of the corresponding output – or more advanced methods of extracting rulesets such as those discussed by Hailesilassie (2016) may significantly improve the understandability and reliability of the explanations.

References

1. Caldwell, S., Tamás, G., Jones, R. and Copeland, L., (2015). 'Imperfect Understandings: A Grounded Theory And Eye Gaze Investigation Of Human Perceptions Of Manipulated And Unmanipulated Digital Images', *Proc. World Congress on Electrical Engineering and Computer Systems and Science.* Barcelona, Spain, July 13-14, 2015.
2. Chattopadhyay, A., Manupriya, P., Sarkar, A., and Balasubramanian, V. N. (2019). Neural Network Attributions: A Causal Perspective. *Proc. 36th International Conference on Machine Learning 97* (Vol. 1, pp. 981-990)
2. Engelbrecht, A. and Viktor, H., 1999. Rule improvement through decision boundary detection using sensitivity analysis. *Lecture Notes in Computer Science*, pp.78-84.
3. Gedeon, T. D., & Turner, S. (1993, October). Explaining student grades predicted by a neural network. In Neural Networks, 1993. IJCNN'93-Nagoya. Proceedings of 1993 International Joint Conference on (Vol. 1, pp. 609-612). IEEE.
4. Hailesilassie, T., 2016. Rule Extraction Algorithm for Deep Neural Networks: A Review. *(IJCSIS) International Journal of Computer Science and Information Security*, 14(7).
5. Zhu, Q. (2020) "Predicted the authenticity of anger through LSTMs and three-layer neural network and explain result by causal index and characteristic input pattern," *3rd ANU Bio-inspired Computing conference (ABCs 2020), paper 83, 7 pages,* Canberra.