Understanding Eye Movements on Mobile Devices by Pruning Redundant Hidden Units

Rui Wang

Research School of Computer Science

Australian National University

Acton ACT 2601 Australia

U6688648@anu.edu.au

Abstract. As data continues to grow exponentially, data processing has increasingly become an important discipline in computer science. Data compression play an important role in data processing as it reduces data transmission time and communication bandwidth, thereby saving a lot of costs. Several previous studies have reported that image compression by pruning redundant hidden units in the neural network is a feasible way to improve the compression rate. However, it is still unknown how the data will be affected by pruning redundant hidden units. In this paper, we use a NEAT (Evolving Neural Networks through Augmenting Topologies) technology to generate a classification neural network to predict the screen size of phones. Then, a data compression neural network, which is a traditional shallow network, is used to compress the dataset. The hidden units of redundant unit pairs that whose separate angle is too small or too large will be pruned in the network. Finally, the NEAT network predicts the phone sizes with compressed dataset. The result of data compression will be evaluated by comparing the accuracy of the classification with original dataset before compression and dataset after compression. It is found that there was no significant difference between the accuracy of the two neural networks.

Keywords: Data Compression, Redundant units, Pruning units, Neural Network, Distinctiveness of Units, Mobile devices, Genetic algorithms, Neuroevolution, Network Topologies.

1 Introduction

Due to the popularity of mobile phones and the development of the Internet, mobile search has become more popular. From September 2013 to August 2014, the use of mobile Internet surged by 67% (Statcounter Global Stats, 2014). However, searching behaviors on the phone is complex than that on the computer because of the novel size of screen. So, copying the interface of a computer search engine is inappropriate. Analyzing the various behaviors of users when searching using mobile phones has become an important research topic. It is believed that understanding the interaction between users and mobile devices will improve the user experience when searching on the phone. (Lagun, Hsieh, Webster, & Navalpakkam, 2014).) Out of the large research efforts devoted to user's interaction when searching on the phone. It has been considered that the screen size of the mobile phone will affect user interaction (Kim, Thomas, Sankaranarayana, Gedeon, & Yoon).

The target of the report is to predict the screen size of the mobile phone by the user's web search behaviors and interaction on mobile devices. So, the mobile phone can automatically display an adapted search engine results page, which will improve the user's search experience. However, the roles of every attribute in the dataset in predicting the screen size is not clear. And a large amount of data will slow down the training speed and reduce the accuracy of prediction. Therefore, compressing data while maintaining data integrity is needed.

2 Method

2.1 Overview

The data compression model is evaluated by the comparison between the accuracy of two classification model before compression and after compression. The flow of the entire report is as follows.

1. Pre-process the whole dataset.
2. Train a model with the pre-processed dataset and calculate the accuracy of classification.
3. Perform data compression to the pre-processed dataset
    - 3.1 Calculate the output activation of the hidden unit.
    - 3.2 Construct unit output activation vectors over the pattern presentation set.
    - 3.3 Calculate angles between every two vectors and Find the angles below 15° and above 165°.
    - 3.4 Delete one unit of the unit pairs and add the weight and bias of the deleted pair to the other unit.
    - 3.5 Repeat step3.4 until all angles are between 15° and 165°.
    - 3.6 Reconstruct the neural network model with remained hidden units.
    - 3.7 Calculate the output of the reconstructed network as the new dataset.
4. Train a model with the compressed dataset and calculate the accuracy of classification.
5. Compare the accuracy of two models.

2.2 Pre-processing

The dataset used in this paper is based on the data collected in Understanding Eye Movements on Mobile Devices for Better Presentation of Search Results (Kim, Thomas, Sankaranarayana, Gedeon, & Yoon, 2016). This 162-row dataset records the search performance and user behaviors during 18 participants searching 9 questions by mobile phone with 3 kinds of screen size respectively. Each row of data has 27 features, including screen size, task number and corresponding search performance and search behavior. The corresponding relationship between research target and collected data is shown in Figure 1. For example, the features that represent search performance are search speed and search accuracy. The data collected to indicate search speed is *Time to first click* and *Task completion duration.*

TABLE 2. Search performance and behavior.

		Mean values			Statistics			
		L	M	S	*p*-value	L	M	S
Search performance								
Search speed	Time to first click [s]	7.70	10.47	9.12	0.195			
	Task completion duration [s]	20.89	24.79	23.08	0.655			
Search accuracy	Correct answer rate [%]	94.44	98.15	94.44	0.589			
Search behavior								
Fixation duration on SERP	Per task [s]	3.97	5.60	5.53	0.087			
	Per link [s]	2.16	1.89	2.49	*	ab	a	b
Clicks	Ranks	1.39	1.52	1.46	0.697			
Scanpath	Minimal scanpath	2.06	2.76	2.26	**	a	b	a
	Compressed sequence	3.33	5.50	4.35	**	a	b	ab
	Compressed minus minimal	1.28	2.74	2.09	*	a	b	ab
Scanning direction	Complete rate [%]	96.30	94.44	100	**	a	a	b
	Linear rate [%]	46.30	31.48	57.41	*	ab	a	b
	Strictly linear rate [%]	11.11	1.85	9.26	0.087			
	Linear/ID rate [%]	81.48	55.56	79.63	**	a	b	a
	Strictly linear/ID rate [%]	46.30	25.93	31.48	*	a	b	a
Skip and regression	Skip [%]	14.81	22.22	7.41	0.087			
	Regression [%]	53.70	74.07	68.52	*	a	b	ab
Scroll	Scrolled rate [%]	3.70	20.37	35.19	***	a	b	b
Trackback	Count	1.07	1.91	1.28	***	a	b	a
Search satisfaction	7-point Likert scale	5.24	4.91	4.20	***	a	a	b

*Significant at 0.05 level. **Significant at 0.01 level. ***Significant at 0.001 level. *Note*. SERP denotes search engine results page, and L, M, and S denotes large, medium, and small, respectively.
Note. Labels a and b indicate the type of result, "a" type is significantly different from "b", but not different to "ab".

Fig. 1. Corresponding relationship between research target and collected data

To train the neural network model, the dataset is randomly split into train set (80%) and test set (20%). With similar effects of cross-validation,10 times shuffle split is applied to improve the reliability of the model.

Preprocessing is applied to the data set in the form of data transformation, data cleansing, and data normalization. Firstly, Categorical attributes like *Screen_size* is converted to numerical data (S -> 0, M -> 1, L -> 2). So, the attribute can be converted into Tensor type in Python. Additionally, some attributes that have been found to have no significant differences in different kinds of screen size has been deleted to accelerate network training. It can be found in Figure 1 that only 12 features are statistically related to the screen size. (Kim, Thomas, Sankaranarayana, Gedeon, & Yoon, 2016). So, attributes like *Time_to_first_click* and *Task_completion_duration* are deleted. Detailed attributes can be seen in the dataset named 'dataset_processed' attached with the report.

2.3 First classification model

2.3.1 Model Selection

To improve the accuracy of predict, a NEAT network and a traditoanl shallow network are compared to find the most suitable model. The traditional shallow network is selected as it shows more stable accuracy even though the accuracy is a bit lower (about 5%) than NEAT network. The design and concept of the both networks are discussed below.

a. NEAT network

The first classification neural network is a network generated from NEAT technology, which is found to have higher average accuracy than traditional shallow neural network. Combining genetic algorithm and evolution strategy, the NEAT technology will leave the neural network with highest accuracy as the final model (Stanley & Miikkulainen, 2002). Firstly, we need to construct a fully connected neural network as the basic model. Then, a genome, which is a species in the nature, is automatically generated to record the information of current network. There is a list of gene nodes and connection genes, in which every node and connection between nodes are encoded with a unique number. The weight and availability of each connection are also included in the list. The details are shown as Figure 2. Then, the weights and structures in the network are changed by mutation. Especially, there are two ways in structure mutations as show in Figure 3. One is adding connection between nodes. The other one is cutting current connection and add a new node in the cutting position. Simultaneously, the genome is updated according to the new network. The next step is crossover between genomes, which is the simulation of reproduction. Additionally, the new genome will be forced to survive for a certain generation. The reason is that the new genome with new structure usually have less fitness (less possibility to survive) compared with genome at the time. However, the new genome may have higher fitness than other

genomes in future, which means it is more suitable in future environment. The rule is significant as it can protect speciation innovation and prevents new species from becoming extinct too soon. The final step in the model is keep the total population at a fixed number during the whole process. To achieve this, we will filter out species that are not suitable for the current environment. In other words, the genomes with less fitness will be deleted from the genome cluster.

The initial number of species is set to 300 and will remain at 150 in further generations. The number of initial hidden units is 15. The generation that genomes will be alive is 20 generations at least. The mutate possibility for weights and connection are 0.7 and 0.05 respectively. These values are set high to increase the diversity of the species. The fitness function in the model is the value of cross entropy as the task of the model is classification. The genome with high value will be considered as not suitable in current environment and will be filter out.

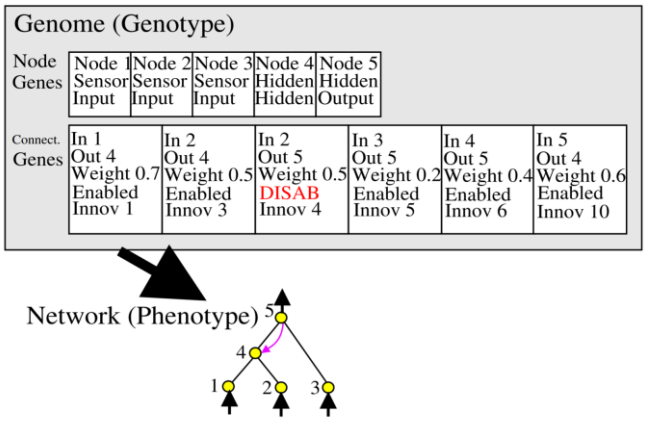

Fig. 2. A genotype to phenotype mapping example

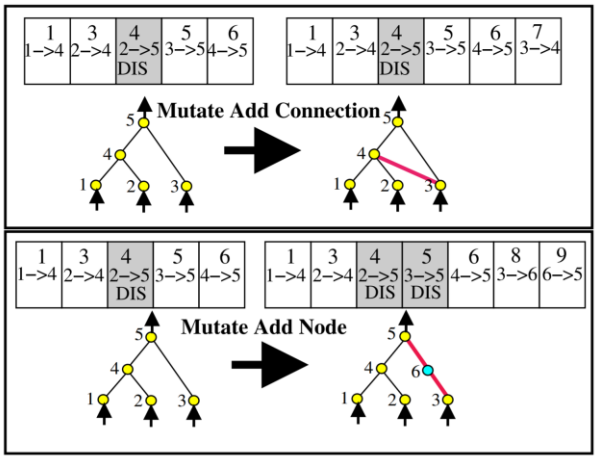

Fig. 3. The two types of structural mutation in NEAT

b. Traditional shallow network

To evaluate the performance of NEAT network, a traditional shallow classification model is used as a control group. The traditional shallow classification model is a 3-layer neural network model. The attribute to predict is *Screen_size* and all other attributes are input features. The number of input units and output units of the model is 14 and 3, respectively, which are the same as the number of input features and the kinds of screen size. The model is found to reach the highest average accuracy with 54.46% when the number of hidden units is 15. The corresponding learning rate and epoch number are 0.01 and 20000 respectively. As the purpose of the model is to classify the screen size, Cross-Entropy is selected as the loss function. To optimize the algorithm, stochastic gradient descent is chosen due to its fast calculation speed.

2.4 Data compression

The data compression network is an auto-associative network with the same number of input and output units. The number of input and output units is equal to the feature number 27. The number of hidden units will determine the degree of compression. In this report, the results of 6, 8, 10, 12, 14 hidden units are discussed below. Mean-square error is applied as all features are numerical. stochastic gradient descent is chosen to accelerate the training process. The sigmoid activation function is used to normalize outputs of hidden units to range 0 to 1. So, the output of all hidden units is expressed in a vector to form a feature map as Figure 4. To use the angular range of 0-180° instead of 0-90°, vector angle calculations are normalized to 0.5 by subtracting 0.5 to the output activation vectors.

	0	1	2	3	4	5	6	7	8	9
0	0.100590	0.295030	-0.367274	-0.017440	-0.429402	0.407550	-0.076092	0.216397	-0.469691	-0.384628
1	-0.180906	-0.291834	-0.031179	0.094670	-0.043910	0.135531	-0.086616	-0.126457	-0.329670	-0.180928
2	-0.243314	-0.273969	-0.012550	0.114688	0.001609	0.128940	-0.075514	-0.122865	-0.331746	-0.177906
3	-0.164184	-0.200488	-0.082329	0.019345	-0.047842	0.278742	-0.235327	-0.040811	-0.307244	-0.267104
4	-0.221723	0.105686	-0.253994	-0.038237	0.072508	0.410790	0.090739	0.179527	-0.102491	-0.261825
...	...	...	...	...	...	...	...	...	...	...
157	-0.218495	-0.100864	0.083393	0.158518	0.085223	0.184793	-0.202984	0.036636	-0.017676	-0.187920
158	-0.068792	-0.152215	0.072285	0.191520	0.049268	0.254699	-0.000742	-0.046079	0.021371	-0.285116
159	-0.199673	-0.076730	-0.033591	-0.054456	0.100306	0.358872	-0.116978	0.131109	0.049927	-0.217455
160	-0.134624	0.041352	-0.084963	0.032971	-0.005230	0.347521	0.031776	0.144693	0.056646	-0.281509
161	-0.166592	-0.058994	0.097192	0.107156	0.076516	0.262012	-0.184538	0.116492	0.058973	-0.220607

Fig. 4. 161 hidden unit activations by pattern

The angles between unit output activation vectors are considered as the distinctiveness of units (Gedeon & Harris, 1991). Unit pairs with an angle that is too large or too small are considered as redundant functionality. One unit of the redundant unit pair is removed, and its weight and bias are added to the other unit in the unit pair. The angles between any two vectors are calculated by the cosine formula and converted into angles as Figure 5. The angles between unit output activation vectors are considered as the similarity of units. Unit pairs with an angle that are too large or too small are considered as redundant functionality. One unit of the redundant unit pair is removed, and its weight and bias are added to the other unit in the unit pair. For example, as shown in Figure 5, the angle between hidden unit 1 and 5 is 8.5° which is below 15°. So, the weight and bias of Hidden unit 1 will be added to Hidden Unit 5 and the Hidden Unit 1 will be deleted.

Pair of Units	Vector Angles	Distinctiveness
1 2	63.8	
1 3	22.9	
1 4	33.3	
1 5	8.5	Similar
1 6	72.8	
1 7	86.6	
2 3	173.2	Complementary
2 4	125.5	
…		
…		
4 5	26.6	
5 6	57.8	
5 7	99.2	
6 7	87.1	

Fig. 5. Angles between two hidden units

2.5 Second classification model

To ensure a single variable and improve the reliability of the model, the structure and weights in both neural networks models are the same. The only difference is that the second model uses compressed dataset as input features rather than that in the original dataset.

3 Results and Discussion

3.1 Classification accuracy & analysis

The structure of NEAT network can be seen in the Figure 6. There is no concept of layer in NEAT network, which is believed to improve the ability of optimizing and complexifying solutions simultaneously (Stanley & Miikkulainen, 2002). The solid and dashed lines represent that the connection is enabled and disabled, respectively. The color and thickness of the line illustrate the size and positive and negative of weight respectively (green: positive, red: negative, thick: large, thin: small).

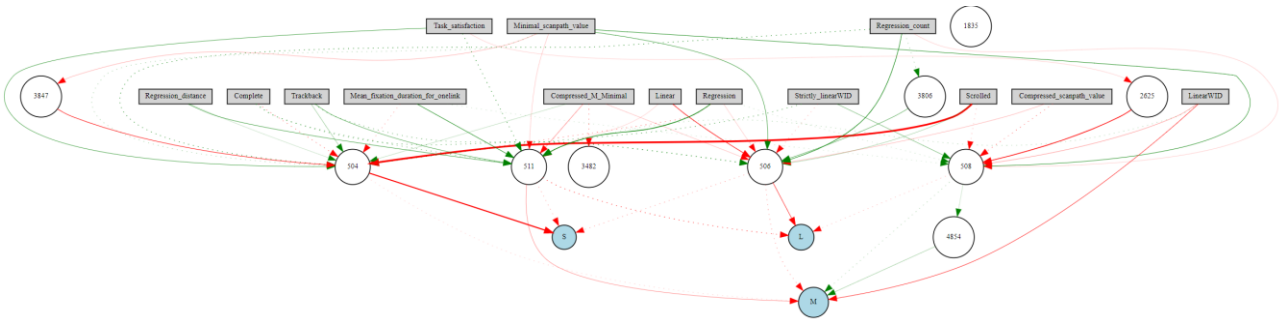

Fig. 6. Angles between two hidden units

The whole process of species competition is shown as Figure 7. It can be found that the number of species decreased to 5 after 50 generations of competition and stabilized at 5 in the further generations. Simultaneously, the best and average fitness rose steadily to -0.43 at the end of training. Through curve average, -1 standard deviation and +1 standard deviation in Figure 8, the fitness of most species were around -0.48 in the whole process.

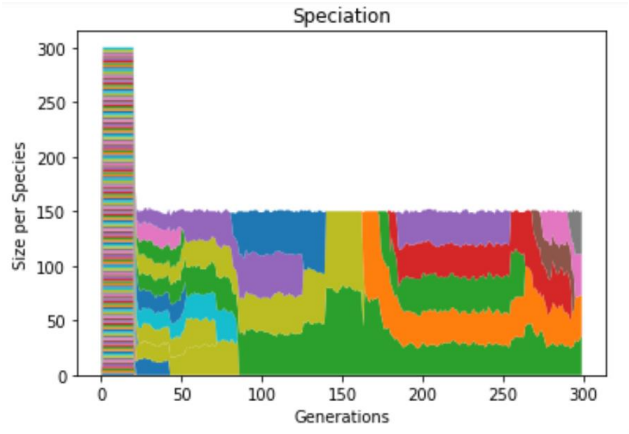

Fig. 7. Trend in species competition

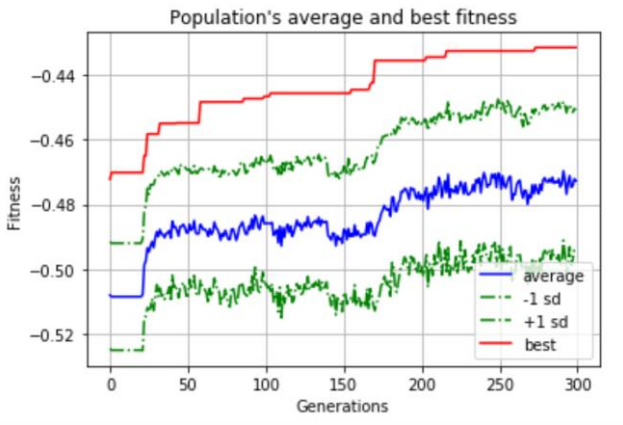

Fig. 8. Population's average and best fitness

Compared with traditional network with 53% accuracy, the NEAT network is proved to have higher accuracy at 58%. Although the optimization process in NEAT technology is slow, the network can jump out of local minimum as it is not limited by the gradient. The problem of time expense can be alleviated by parallel computing power. Another drawback of the NEAT technology is that the accuracy is not as stable as traditional network as the uncertainty of mutation. A sufficiently large population is a feasible solution.

3.2 Data compression result & analysis

The comparison of different initial hidden units is shown in Table 1. It is notable that the number of similar pairs increased with the increase of initial hidden units. The pairs of similar units are considered as redundant functionality and should be removed (Gedeon & Harris,1991). In other words, when there are more initial hidden units, the function of the unit becomes more likely to be redundant. These redundant hidden units will not significantly improve the performance of the model but reduce the training speed of the model. Therefore, these redundant units should be deleted to improve speed of learning. Meanwhile, the number of initial hidden units should be set to an appropriate value to avoid redundancy.

Table1. Number of redundant pairs with different hidden units

Initial Hidden units	6	8	10	12	14
Pairs of similar units	0.5	0.8	1.2	1.6	3.2

3.3 Classification result & analysis

The comparison of the classification result is shown in Table 2. Firstly, it is noted that the overall average accuracy is low. One possible reason is the insufficient sample size. A large amount of data is needed to train a qualified neural network model (D'souza, Huang & Yeh,2020). It is also possible that the connection between features and targets is weak. Additionally, the average accuracy of the neural network with the compressed dataset is less than the average accuracy of the neural network but the difference between them is acceptable. The average accuracy of the model decreased by about 5%, which means that the main functionality of features is preserved. A reasonable explanation is that redundant hidden units are deleted, and the weights of the hidden units which contains principal component is relatively increased. Furthermore, the average accuracy of both models decreased with the increase of hidden units. One possible reason is overfitting due to too many hidden units.

Table 2. Accuracy with datasets before compression and after compression

Hidden units	6	8	10	12	14
Average accuracy before compression	51.52%	51.18%	49.84%	53.56%	45.45%
Average accuracy after compression	50.36%	51.16%	48.48%	51.52%	42.44%

4 Conclusion and Future Work

In conclusion, the result discussed before proved the feasibility of NEAT technology and data compression technology. The network generated by NEAT technology is proved to predict the screen size more accurately. The hidden unit output vector over the pattern presentation set determines the distinctiveness of hidden units. Removing similar and complementary units can reduce the redundancy of neural network models. Compared with the previous study, the report declares a new data compression technology, which is pruning redundant hidden units. The new technology can be applied to various data processing. It is also confirmed that although the connection is not obvious significantly, the screen size has a certain relationship with user search behavior. More data and experiment are needed to confirm the view. The shortcoming of this report has been the failure to study the combined behaviors of three or more hidden units. Although this situation is rarely negligible, the redundancy of the neural network model can be further reduced if combined behaviors can be removed. This can be a further development direction in the future.

Reference

1. D'souza, R.N., Huang, P. & Yeh, F. Structural Analysis and Optimization of Convolutional Neural Networks with a Small Sample Size. *Sci Rep* **10,** 834 (2020). https://doi.org/10.1038/s41598-020-57866-2
2. Gedeon, T. D., & Harris, D. (1991). Network reduction techniques. In *Proceedings International Conference on Neural Networks Methodologies and Applications* (Vol. 1, pp. 119-126).
3. Gedeon, T. D., & Harris, D. (1992, June). Progressive image compression. In *Neural Networks, 1992. IJCNN., International Joint Conference on* (Vol. 4, pp. 403-407). IEEE.
4. Kim, J., Thomas, P., Sankaranarayana, R., Gedeon, T., & Yoon, H. J. (2016). Understanding eye movements on mobile devices for better presentation of search results. *Journal of the Association for Information Science and Technology*, 67(11), 2607-2619.
5. Lagun, D., Hsieh, C.-H., Webster, D., & Navalpakkam, V. (2014). Towards better measurement of attention and satisfaction in mobile search. *In Proceedings of the 37th international ACM SIGIR conference on research & development in information retrieval* (pp. 113–122).
6. Stanley, K. O. and Miikkulainen, R. (2002). "Efficient evolution of neural network topologies," *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Honolulu, HI, USA, 2002, pp. 1757-1762 vol.2, doi: 10.1109/CEC.2002.1004508.
7. Statcounter Global Stats. (2014). *Mobile internet usage soars by 67%.* Retrieved from http://gs.statcounter.com/press/mobile-internet-usagesoars-by-67-perc