QAO_REU_NN_Data_Sheet

Gage Erwin

August 2023

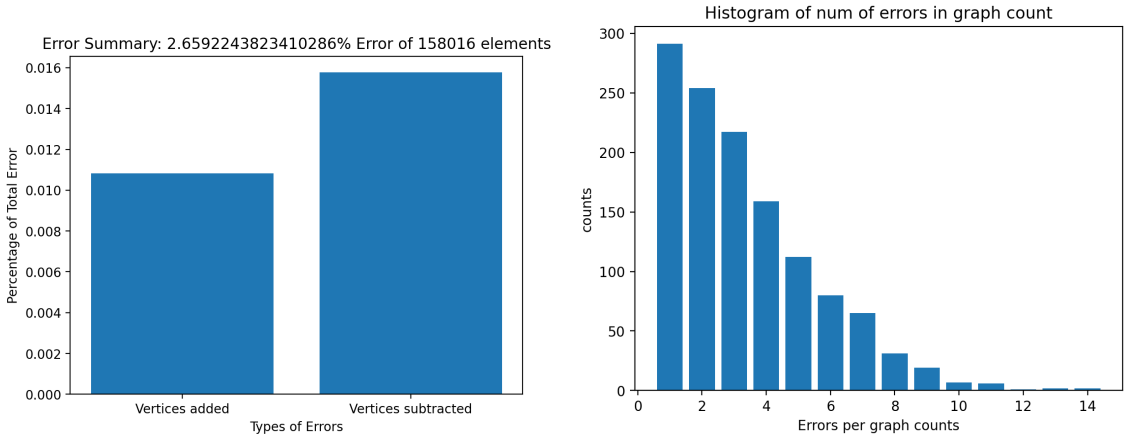
This data sheet is given as a way to lead the reader through the accuracies and models that gage presents in his github. This sheet will not give too many details and it is up to the reader to explore the code to understand how it works and fix any issues, if there are any, for further research. If there are any missunderstandings please reach out.

0.1 NN for predicting graphical structure

- The file that will be used in this will be called "NNQAOA3_8vertices.ipynb"
- In this file, I take the qaoa output of graphs in "graph8.g6" where I produce the data in the "QAOAREU_NN_Data_Production.ipynb".
- Data was produced with 16384 counts and there are 3 sets $p = 1, p = 2$, and $p = 3$, labeled "QAOACountsAugust1_8_16384_Counts_1P.json", "QAOACountsAugust1_8_16384_Counts_2P.json", and "QAOACountsAugust1_8_16384_Counts_3P.json" respectively.
- The accuracies produced by such data are presented in Table 1.
- Gage's poster is also presented in the GitHub with label "QAOA_Research_Poster_Erwin-5", however, it should be noted that data in the poster was produced with 1024 counts at $p = 1$ from "QAOACountsJuly7_8.pkl". It is possible to recreate, but I do not present the downloaded model.
- I also present the figures of their training after the table, leaving it up to the reader to make claims to test with the code.

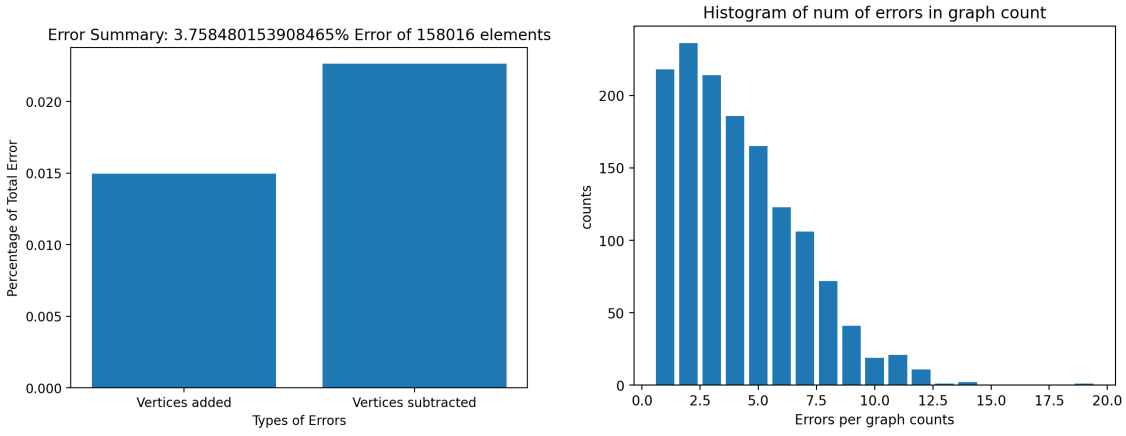
Model Name	hcnet	hcnet	hcnet
Model File	QAOACounts_16384_8_1P	*_2P	*_3P
Data File	hcnet_unconnected_8_16384_16_1P	*_2P	*_3P
Element accuracy (Training/Testing)	99%/97%	96%/96%	95%/95%
Graph accuracy(Training/Testing)	70%/50%	45%/43%	39%/39%

Table 1: Table of accuracies for the NN predicting Graphs from QAOA output after training for 1000 epochs, learning rate of .001, and batch size of 16. All of these were trained on the same architecture HCNET on NNQAOA3_8vertices



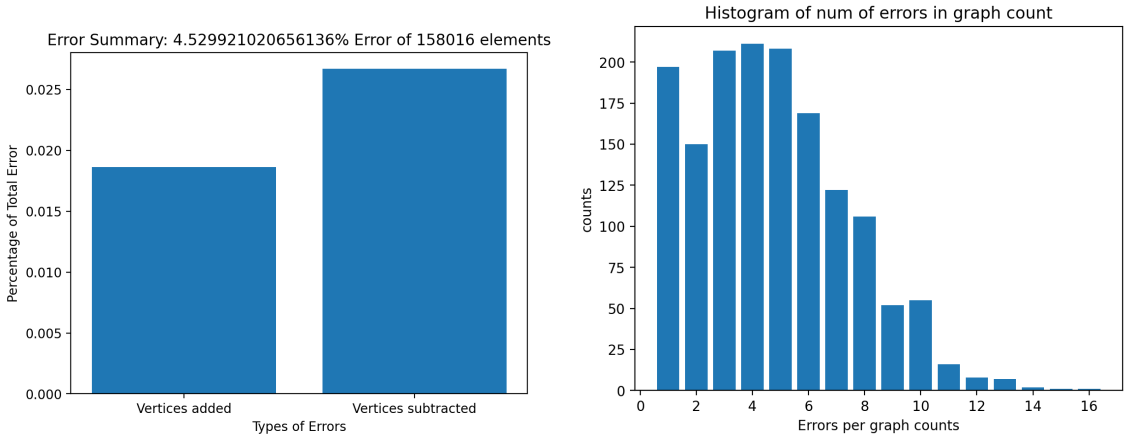
(a) (b)

Figure 1: Error Analysis of QAOACounts_16384_8_1P. Graph (a) presents the number of elements added/subtracted in the predicted adjacency graph that were not present in the true graph. Graph (b) presents a analysis of how many errors per graph were present. These two graphs show as a method to test how well the models are doing and what types of errors are causing lower accuracies.



(a) (b)

Figure 2: Error Analysis of QAOACounts_16384_8_2P.Graph (a) presents the number of elements added/subtracted in the predicted adjacency graph that were not present in the true graph. Graph (b) presents a analysis of how many errors per graph were present. These two graphs show as a method to test how well the models are doing and what types of errors are causing lower accuracies.



(a) (b)

Figure 3: Error Analysis of QAOACounts_16384_8_3P. Graph (a) presents the number of elements added/subtracted in the predicted adjacency graph that were not present in the true graph. Graph (b) presents a analysis of how many errors per graph were present. These two graphs show as a method to test how well the models are doing and what types of errors are causing lower accuracies.

0.2 NN for predicting Optimized

- Here I present a NN model that either takes in the QAOA output, a graph, or both, and outputs either the maximum cut in binary or the binary solution of the graph for the maximum cut.
- The file that will be used in this will be called "NNQAOA_4.ipynb"
- The data file used to train and test this model is "QAOACountsJuly7_8.pkl" with 80% of the elements used to traing and rest to test.
- Accuracies for this a presented in Table 2

Model File	Input/Output	Training Accuracy	Testing Accuracy
GraphtoDigit.pth	Graph/Maximum Cut	85%	82%
GraphtoBin.pth	Graph/Binary Solution	51%	50%
ExptoDigit.pth	QAOA Dist./Maximum Cut	44%	35%
ExptoBin.pth	QAOA Dist./Binary Solution	37%	30%
BothtoDigit.pth	Both/Maximum Cut	80%	77%
BothtoBin.pth	Both/Binary Solution	67%	51%

Table 2: Table of accuracies for the NN predicting Graphs from QAOA output after training with a learning rate of .001, and batch size of 256.