TensorFlow Evaluation

The TensorFlow model evaluated throughout this analysis uses the MNIST dataset, a collection of over 70,000 images of handwritten numbers ranging from zero to nine. The evaluation aims to build and train the model with the MNIST dataset to extract some metric data of the operation, accuracy, and optimization performance based on parameter changes that could affect the model's overall performance. Furthermore, two versions of the same were created version 1 with the original parameters and version 2 with the modifications for comparison.

1. **Accuracy of the Model:** The overall accuracy of the model after each step (epoch) was 95% of accuracy after completing the whole twenty epochs, which was achieved with the original configuration of the model of 512 hidden neurons, one hidden layer, and a learning rate of 0.5, that leaves the other five percent of misclassified images.
2. **Misclassified Images:** Some of the misclassified images range as follows; EPOCH..1 label: {3, 2, 9, 3, 3}, EPOCH..2 labels: {9, 2, 3, 2, 3}, EPOCH..3 labels: {9, 4, 2, 3, 6}, EPOCH..4 labels: {3, 9, 4, 2, 3}, EPOCH..5 labels: {9, 4, 2, 3, 6}, EPOCH..6 labels: {9, 4, 2, 3, 6}, EPOCH..7 labels: {9, 4, 2, 3, 6}, EPOCH..8 labels: {9, 4, 2, 3, 6}, EPOCH..9 labels: {9, 4, 2, 3, 6}, EPOCH..10 labels: {9, 2, 3, 6, 9}, EPOCH..11 labels: {9, 4, 2, 3, 6}, EPOCH..12 labels: {9, 4, 2, 3, 6}, EPOCH..13 labels: {9, 4, 2, 3, 6}, EPOCH..14 labels: {9, 4, 2, 3, 6}, EPOCH..15 labels: {9, 4, 2, 3, 6}, EPOCH..16 labels: {9, 4, 2, 3, 6}, EPOCH..17 labels: {9, 4, 2, 3, 6}, EPOCH..18 labels: {9, 4, 2, 3, 6}, EPOCH..5 labels: {9, 4, 2, 3, 6}, EPOCH..19 labels: {9, 4, 2, 3, 6}, EPOCH..20 labels: {9, 4, 2, 3, 6}.

3. **Increasing/Decreasing the Hidden Neurons:** The number of hidden neurons was increased from 512 to 712 and then to 950, which did not affect the accuracy at all, resulting in the same 95% accuracy. However, despite decreasing the number of neurons from 512 to 350, 150, and 100, the model's accuracy remains unchanged, but the processing speed improves slightly.
4. **Learning Rates:** Thought that four different learning rates were implemented to gauge how affected, ranging from 0.1 to 0.0001. It was found that when applying a learning rate of 0.1, the accuracy ranges from 94 to 95 overall accuracy; similar results were observed when applying the learning rates of 0.01 and 0.0001; the overall accuracy in both cases was 95%.
5. **Adding more Hidden Layers:** when adding another hidden layer to the model, it was observed that the overall accuracy dropped exponentially; the model's accuracy dropped by 85%, scoring an overall accuracy of 10%, which was underwhelming.
6. **Batch Sizes:** Both algorithm versions have tested three different batch sizes. The results have shown that a batch size of 50 with a learning rate of 0.5 in version 1 of the algorithm did not affect the accuracy retaining of 95%. Whereas a batch size of 200 in the same code reduced the accuracy by 2%, giving an accuracy of 93%; the same results were shown when the batch size was increased this time to 500, losing a 3% accuracy and giving an overall accuracy of 92%. However, version 2 of the code was different; accuracy remained the same despite applying different learning rates ranging from 0.1 to 0.0001 and increasing the batch sizes from 50 to 500; the overall accuracy remained at 10%.

Overall accuracy from the multi-layer perceptron

Based on the analysis and the results of this research, the best accuracy that one can get out of this model shall depend on the configuration given to the model. The best accuracy achieved throughout the research was ~ 94 to 95%, which was pretty good. However, such accuracy was with the original parameters of the model, for instance, a learning rate of 0.5, a batch size of 100, and one hidden layer—the accuracy dropped exponentially once a second layer was introduced to the algorithm.

Conclusion

The research has shown not only the level of accuracy that a TensorFlow model could achieve based on the task for which such a model is created but how different configurations can impact the model's accuracy. Finally, such impact could affect the model both ways, positive or negative, based on the parameters and the problem that the algorithm is set to solve.