Critical Thinking Module 4 Option 1

Benjamin Gutierrez

Colorado State University Global

CSC 580: Applying Machine Learning and Neural Networks - Capstone

Dr. Lori Farr

January 29, 2023

The model is a neural network that has been trained to categorize molecular data. To do this, it uses a Tox21 dataset and some TensorFlow algorithms. To avoid overfitting, the network contains a single hidden layer with 50 neurons, an output layer, and a dropout layer. The final model obtained a validation accuracy of 91.5% after being trained for 10 epochs as shown by Figure 2.

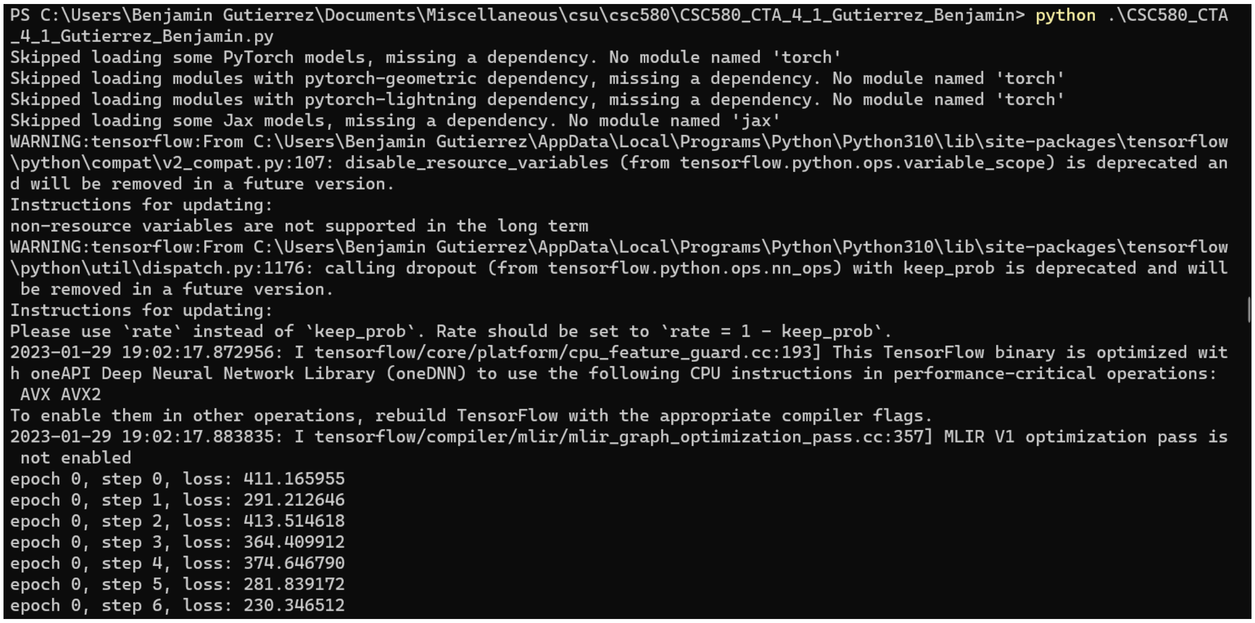
The code loads the Tox21 dataset and divides it into three parts: training, validation, and testing. It then creates a hidden layer with ReLU activation and specifies placeholders for the inputs and targets. The output layer, which utilizes a sigmoid activation function to output class probabilities, completes the design. Adam optimization and cross-entropy loss are used to train the model.

The model's design is straightforward, which might be a benefit in terms of interpretability and training speed. The use of merely one hidden layer, on the other hand, may restrict the model's capacity to capture complicated connections in the data. The dropout layer is used to mitigate the danger of overfitting, which occurs when the model memorizes the training data but fails to generalize to new data.

The model's validation accuracy of 91.5% indicates that it has learnt a meaningful association between molecular data and toxicity categorization. However, the accuracy score alone is insufficient to adequately assess the model's performance.

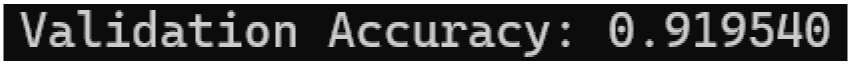
Finally, the TensorFlow code illustrates a basic neural network architecture trained on molecular data for toxicity classification. After 10 epochs of training, the model attained a validation accuracy of 91.5%, which is a respectable result. To acquire a thorough picture of the model's performance, other assessment measures should be evaluated.

Figures



```
PS C:\Users\Benjamin Gutierrez\Documents\Miscellaneous\csu\csc580\CSC580_CTA_4_1_Gutierrez_Benjamin> python .\CSC580_CTA
_4_1_Gutierrez_Benjamin.py
Skipped loading some PyTorch models, missing a dependency. No module named 'torch'
Skipped loading modules with pytorch-geometric dependency, missing a dependency. No module named 'torch'
Skipped loading modules with pytorch-lightning dependency, missing a dependency. No module named 'torch'
Skipped loading some Jax models, missing a dependency. No module named 'jax'
WARNING:tensorflow:From C:\Users\Benjamin Gutierrez\AppData\Local\Programs\Python\Python310\lib\site-packages\tensorflow
\python\compat\v2_compat.py:107: disable_resource_variables (from tensorflow.python.ops.variable_scope) is deprecated an
d will be removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
WARNING:tensorflow:From C:\Users\Benjamin Gutierrez\AppData\Local\Programs\Python\Python310\lib\site-packages\tensorflow
\python\util\dispatch.py:1176: calling dropout (from tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will
 be removed in a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
2023-01-29 19:02:17.872956: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized wit
h oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations:
 AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2023-01-29 19:02:17.883835: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:357] MLIR V1 optimization pass is
 not enabled
epoch 0, step 0, loss: 411.165955
epoch 0, step 1, loss: 291.212646
epoch 0, step 2, loss: 413.514618
epoch 0, step 3, loss: 364.409912
epoch 0, step 4, loss: 374.646790
epoch 0, step 5, loss: 281.839172
epoch 0, step 6, loss: 230.346512
```

Figure 1: Running the model



```
Validation Accuracy: 0.919540
```

Figure 2: Validation Accuracy after 10 epochs

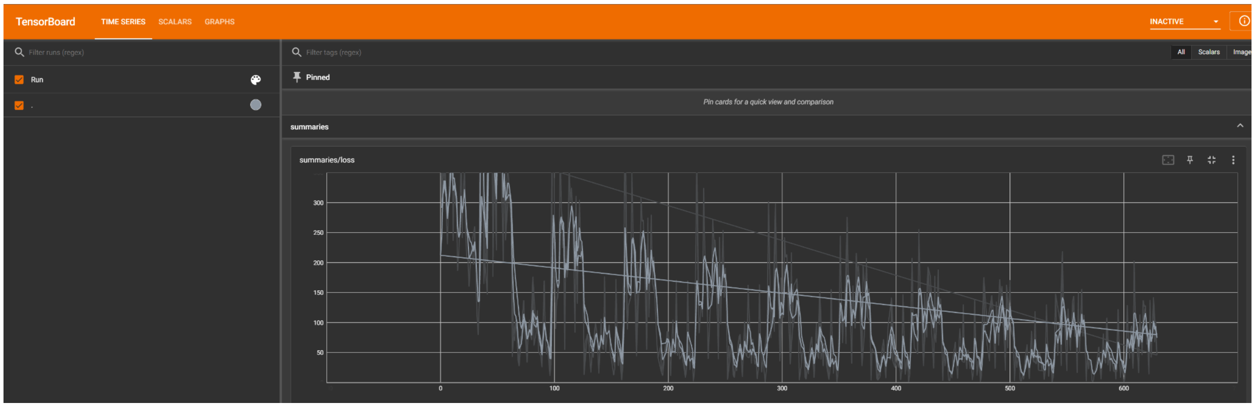


Figure 3: TensorBoard graph for the model

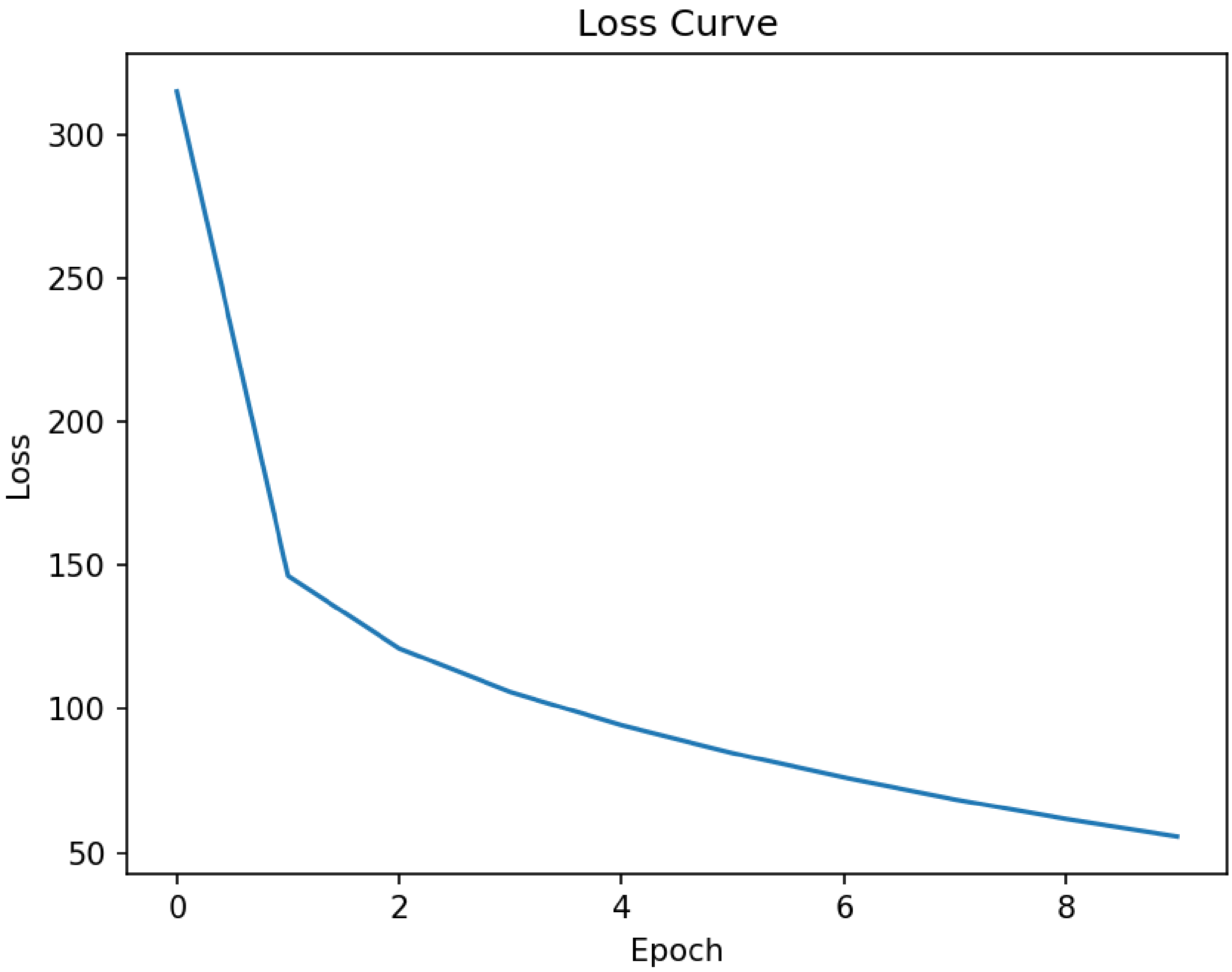


Figure 4: The loss curve