CS583A: Course Project

Zubair Shaikh

December 5, 2019

1 Summary

In this competition, we developed a models to predict the metered building energy usage in the following areas: chilled water, electric, hot water, and steam meters. The data comes from over 1,000 buildings over a three-year time-frame. With better estimates of these energy-saving investments, large scale investors and financial institutions will be more inclined to invest in this area to enable progress in building efficiencies. This was an active competition.

The final model we choose is Deep Learning model with multiple Dense layers and embedding layers for the input data, along with Dropout layers and batch normalization layers.

We implement the neural network using Keras and ran the code on a Google Colab with GPU enabled.

The evaluation metric for this competition is Root Mean Squared Logarithmic Error.

In the public leader board, my score is 1.16; we rank 1791 among the 2929 teams. The private leaderboard is not available until December 19, 2019.

2 Problem Description

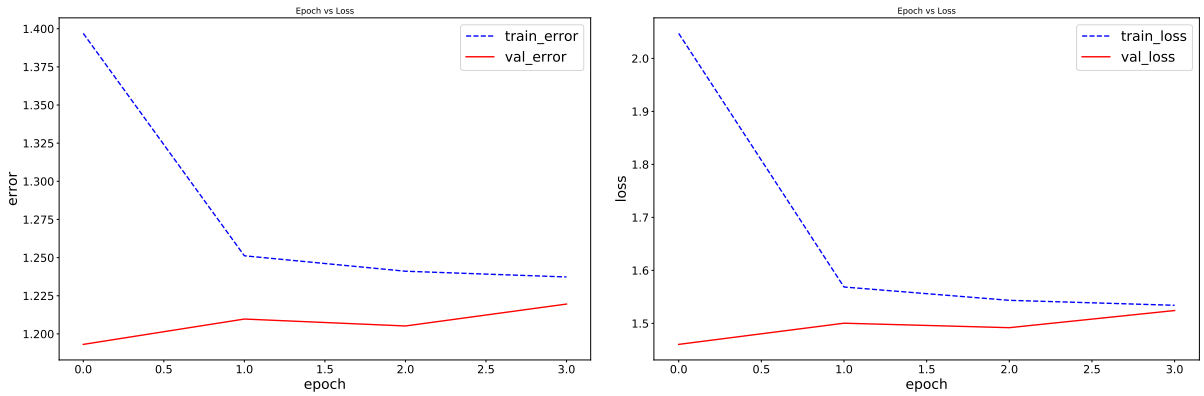
Problem. The problem is to predict the meter reading for a building on a certain hour given the values of weather conditions and the building conditions. This is a time-series and prediction problem. The competition is at `https://www.kaggle.com/c/ashrae-energy-prediction`.

Data. The data are . The number of training samples is $n = 19,777,045$. The number of predictions to makes is $n = 41,697,600$. All of them in one file to make the time-series prediction easier.

Challenges. The size of the data was a huge challenge. Loading and reducing its size to make it available to train made was a difficult part of the problem. We used data conversion and type casting to reduce the space by 46%

3 Solution

Model. The model is a neural network with Input Layer for all the input features passed through respective embedding layers and flattened and concatenated together and the passed through Dense, Dropout, Batch Normalization Layer, then again through Dense, Dropout and some more features



(a) The Error on the training set and validation set.

(b) The loss on the training set and validation set.

Figure 1: The convergence curves.

passed through Input Layer Concatenated together and passed through Dense, Dropout, Batch Normalization, Dense, Dropout and the output Dense layer.

The upper model features are passed through Dense layers and the lower model features are attached to them using the Input Layers.

Implementation. We used Keras to develop the model using different layers. Our code is available at `https://github.com/compmonk/CS583/tree/master/FinalProject`. We run the code on a Google Colab and it takes 1.2 hours to train the model.

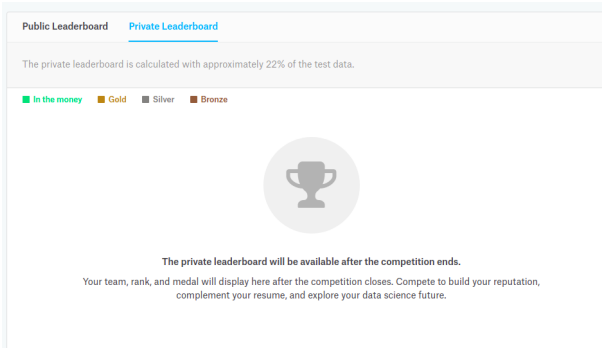
Settings. The loss function is Mean Square Error The optimizer is Adam with learning$_rate$ =0.001.$dropoutlayers$ 0.1, 0.1.$epochs$ =10.$folds$ =2$forKFoldValidation.batchsize$ =1024.

Advanced tricks. Batchnormalization

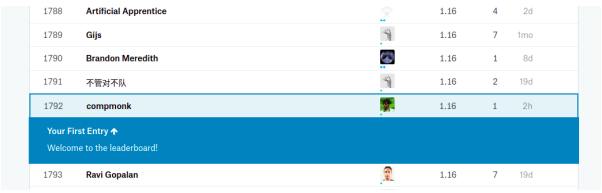
Cross-validation. We used 2-Fold cross validation

4 Outcome

We participated in an active competition. Our score is 1.16 in the public leader board. We rank 1791/2929 in the public leader board and the private leader board is not available till December 19, 2019 The screenshots are in Figure 2.



(a) Private leaderboard.



(b) Public leaderboard.

Figure 2: Our rankings in the leaderboard.