ASSIGNMENT-3 REPORT

VAISHNAVI PENTA
VISHAL SAGAR KARRI

In our analysis of time series data, we analyzed 14 different models. We started with a baseline model based on common-sense procedures that had a Mean Absolute Error (MAE) of 2.62. Following that, we created a minimal machine learning model with a dense layer, which yielded a somewhat higher MAE of 2.70. However, the dense layer model performed poorly due to levelling of the time series data, which damaged its temporal context. We also tried a convolutional model, but it produced poor results because it treated all data segments identically, even after pooling, disturbing the sequential order of the data.

Recognizing the usefulness of Recurrent Neural Networks (RNNs) for time series data, we investigated their potential. RNNs have the ability to incorporate knowledge from previous steps into their present decision-making process, allowing them to detect dependencies and patterns in sequential data. However, the basic Simple RNN consistently performed poorly across all models because to its inability to deal with the "vanishing gradient problem" in deep networks. This issue makes the network difficult to train efficiently.

To overcome this difficulty, we used more complex RNN variations such Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). Our experiments demonstrated that the simple GRU model outperformed other models because it can capture long-range dependencies in sequential data while being more computationally economical than LSTMs.

We then tested six different LSTM models with differing units in stacking recurrent layers (8, 16, and 32), with the 8-unit model outperforming the others. We also used recurrent dropout to avoid overfitting and tried bidirectional data presentation to improve accuracy and address the forgetting problem. These LSTM models consistently produced lower MAE values than the common-sense model.

We then investigated six different LSTM models with differing units in stacking recurrent layers (8, 16, and 32), with the 8-unit model outperforming the others. We also used recurrent dropout to avoid overfitting and experimented with bidirectional data presentation to improve accuracy and address the forgetting issue. These LSTM models consistently had lower MAE values than the common-sense model.

A higher MAE of 3.79 was obtained by combining a 1D convolution model with an RNN, most likely as a result of the convolution's inability to maintain information order. For time series analysis, we therefore advise against using basic RNNs and instead to use more sophisticated RNN designs like LSTM and GRU, which are made especially to handle situations like these. Although LSTM is frequently preferred for time series data, our research indicates that GRU might provide more effective outcomes. Hyperparameters including the number of units in stacked recurrent layers, recurrent dropout rates, and the utilization of bidirectional data presentation should all be fine-tuned in order to maximize GRU models.

Additionally, it is advisable to give priority to RNN architectures designed for sequential data, since 1D convolution with RNN did not produce results that were sufficient. Convolutional algorithms are less appropriate for time series data processing since they have a tendency to mess with the information's order.