**ADVANCED MACHINE LEARNING**

**Assignment – 3**

**Chinthakindi Nithin Kumar - 811289329**

**1. INTRODUCTION**

In this project we are dealing with the optimal time series project that is the weather forecasting, which is summed up to the weather prediction of the future using historical data and applying a various range of modeling methods and techniques. A complex and sophisticated task that entails the correct data predecuding, model choice and evaluation components to generate reliable forecasts. Each observation (data point) has several features (variables) like the temperature, precipitation, humidity, wind speed, etc... In a specific place and time. In this way, the data progresses in a sequential order forming a series of time, which each reading depends on the reading obtained earlier.

Data processing which, by and large, contains elements of data cleaning, normalization, feature engineering, and defining a split point between train, validation, and test datasets.

The invention of machine learning has seen the development of the models such as the recurrent neural networks (RNNs), long short-term memory networks (LSTMs), gated recurrent units (GRUs), and convolutional neural networks (CNNs) becoming more competent in successful results of weather forecasting. In this project we have implemented a few to check the prediction.

Parameter optimization of the model will be accomplished by calculation of a loss function (e.g., mean squared error, mean absolute error) and using gradient descent (a basic method to find a minimum for a given function) as an optimization technique. Models can be trained with historical weather statistics, and their performance can be evaluated by means of metric like Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), on verification data as well. Proper setting values of the hyperparameters and model selection is critical for optimal operation. In this project, we have referred to the MAE to check how well a model accuracy is.

**2. METHODOLOGY**

* The dataset is prepared through standardizing their features, and then it is divided into training, validation, and test sets. Sequencing is performed on the time-series data and then used to feed RNN models.
* Employing an RNN helps to alleviate the hidden units in the recurrent layers by adjusting their number for the sake of improving performance.
* Through LSTM (Long Short-Term Memory) models, we can check if the model performance is working well, which is the result of the LSTM cells that can model long-term relationships. Contrasts are drawn with basic RNN to realize if there are any more significant improvements.
* The last model we used is the 1D Convolutional Neural Network (1D CNN) + RNN: The use of 1D CNN layer in combination with RNN layer. The CNN Layer retrieves feature from the input series, and these are given to RNN for temporal processing.
* Every model is trained and evaluated either the training set or validation set. The performance measurement of MAE is done to compare and study the impact of different algorithms on the models, and we used rmsprop optimizer for learning purposes. To achieve the goal of temperature prediction we have chosen to use MAE because this statistic can be well applied to continuous numerical values predictions.

**3. RESULTS**

|  |  |  |
| --- | --- | --- |
| Model Type | Validation MAE | Test MAE |
| Common-sense | 2.44 | 2.62 |
| Basic ML Model | 2.62 | 2.71 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model Type | No. of Dense units | Loss value | Test MAE | Validation MAE |
| Basic ML Model | 8 | 10.6583 | 2.67 | 2.57 |
| Basic ML Model | 16 | 13.1117 | 2.66 | 2.87 |
| Basic ML Model | 32 | 11.4999 | 2.67 | 2.69 |
| Basic ML Model | 64 | 11.3549 | 2.67 | 2.55 |
| 1D Convolution Model | 16 | 16.3099 | 3.20 | 3.22 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RNN | No. of Dense Units | Dropout | Loss Value | Test MAE | Validation MAE |
| LSTM | 16 | NO | 10.6071 | 2.55 | 2.37 |
| LSTM | 16 | YES | 11.0949 | 2.62 | 2.42 |
| GRU | 16 | YES | 9.8169 | 2.46 | 2.34 |
| Bidirectional LSTM | 16 | NO | 10.8788 | 2.60 | 2.52 |

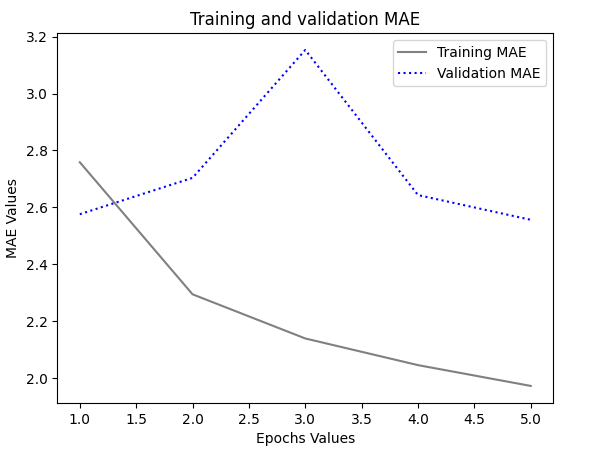
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Combination of 1D Convent and RNN | No. of Dense Units | Dropout | Loss Value | Test MAE | Validation MAE |
| 1D + RNN | 16 | YES | 10.9585 | 2.60 | 2.44 |

**4. SUMMARY**

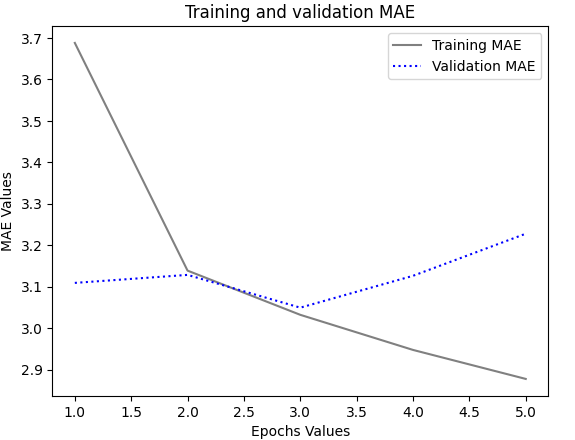
* The number of densely populated units is directly proportional to the rate of the loss indicator, but it is not in a consistent fashion improved MAE. Similarly, for instance, using 8 thick layers, model drops the validation MAE from 2.57 to 2.55, which when using 64 dense layers, it achieves. But the 9-unit density hasn't improved the input.
* LSTM 16 dense units and no dropout show outperformance among RNN types post the fact that has Validation MAE checks in at 2.37 and Test MAE at 2.55. Dropout in the model causes generalization error and thus overall, MAE to grow. Bi-directional LSTM was better with a validation MAE of 2.52 and a test MAE of 2.60 when used, whereas uni-directional LSTM yielded the worst results with a validation MAE of 3.32 and a test MAE of 3.20.
* On the other hand, we have confirmed the 16-unit layout for the 1D Convolutional model and the other RNN models (LSTM layer, GRU and Bidirectional LSTM) by giving the Test MAE as well as loss function.
* Have made LSTM model consisting of dropout (0.5) and 1d\_convnet in conjunction with RNN since such a combination obtained one of the best MAE. The hybridization showed a MAE OF 2.60 and a loss of function of 10.9585 that is the highest among all the model performances. This said the model is likely to give accurate predictions because the MAE and loss function are low.
* Even though LSTM RNNs do not show the best results in weather time-series forecasting, they still come ahead of all other models. Also, the hybrid architecture comprising of a combination of 1D convolutional layers and RNNs is promising, implying a diversification in research towards finding other ways of improving accuracy. These research results show that picking suitable values of neural network architectures and methodology of analysis that matches the quality of the time-series data are of the essence.

**Attaching few plots showing the Training MAE and Validation MAE**

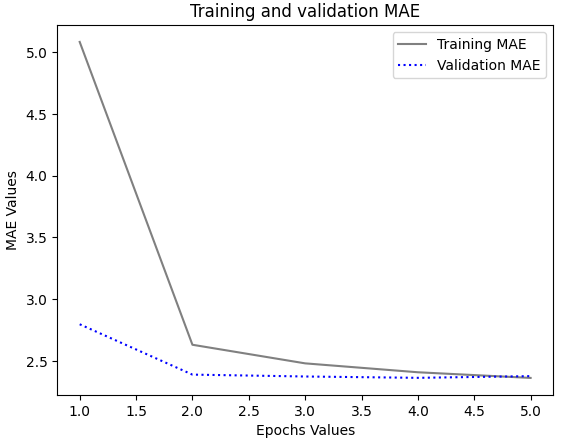
1. A Basic Machine Learning Model



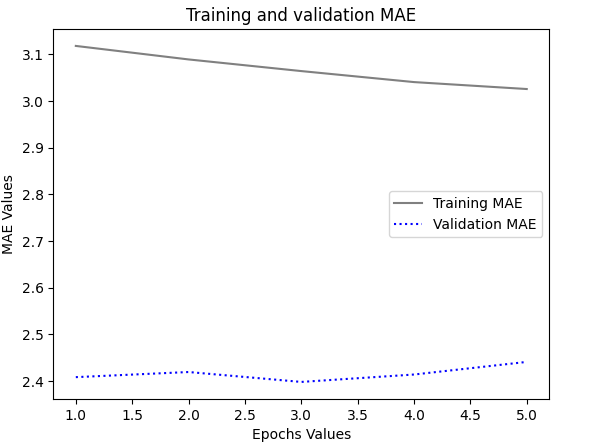
1. 1D Convolution Model



1. LSTM Model



1. Combination of 1D Convent and RNN



**5. SUGGESTIONS**

* Applying MAE (Mean Absolute Error) to time-series data is relevant, especially when the target is a prediction of a continuous value as for example, it is the temperature.
* A neural network with recurrent LSTM model showed better results vs a more common ML and convolutional model with a direction into the time dimension. Such infrastructures, being now, easily allow for time sequences detection in time-series data.
* Additionally, we can configure the hyperparameters of the LSTM models like selecting the number of units, dropout rates, and the learning rate to make the performance better without overfitting which affects greatly in deeper networks or networks that has large number of parameters.
* Besides MAE, on can also use Mean Squared Error (MSE), Root Mean Squared Error (RMSE), among other evaluation metrics to get a thorough overview and complete the model robustness assessment.
* Further deed may include normalization of feature scales or differencing the data to stabilize the time-series dynamics variance.

**6. CONCLUSION**

Among various types of models that are examined to cater to the time-series weather forecasting purpose, RNN architectures particularly LSTM models outperform both simple machine learning and 1D ConvNets. The combination of proper hyperparameter tuning methods, and regularization techniques leads to better models which in turn enhance the forecasting accuracy thanks to considering the best setting for the model. Moreover, the incorporation of 1D-convolution layers with RNNs could manifest opportunities for using spatial and temporal dependencies in data sets. Furthermore, these results highlight the need for picking up neural network architectures and methods adapted to time-series features. In addition, they prove us to go a step ahead and explore more for future improvement of the forecasting models applied to weather prediction.