**Time- Series Data Report**

**Introduction:**

This assignment focuses on using various deep learning models to forecast temperature using the Jena weather dataset. The dataset contains 15 weather-related parameters and comprises a significant 420,451 samples. The main goal is to explore and compare the effectiveness of different neural network architectures for time series forecasting.

A total of 14 models were developed to tackle the time-series forecasting task. Initially, I established a non-machine learning baseline model, which yielded a Mean Absolute Error (MAE) value for test data is 2.62. Following that, I constructed a basic machine learning model employing dense layers, resulting in a slightly higher MAE of 2.66. However, this model encountered challenges as it flattened the time series, thereby losing the temporal context inherent in the original data. Consequently, it performed poorly when adapted to a convolutional model, as pooling operations treated all data segments equally, disrupting the temporal sequence of information. Recognizing the importance of preserving time information, I pivoted to Recurrent Neural Networks (RNNs), specifically designed for handling time series data.

Recurrent Neural Networks (RNNs) possess a distinctive ability to incorporate information from previous time steps into current decision-making processes, facilitating the capture of intricate dependencies and patterns within sequential data. The internal state of an RNN serves as a memory of past inputs, allowing the model to effectively model sequences of varying lengths. However, the Simple RNN architecture, although theoretically capable of retaining information from all preceding time periods, encounters practical challenges. Notably, it is susceptible to the vanishing gradient problem, which hinders its training, particularly in deep networks. Furthermore, empirical observations from the graph revealed that the Simple RNN emerged as the poorest performer among all models tested. To address this limitation, I leveraged Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) architectures, both available as part of the Kera’s library, as they are designed to mitigate the vanishing gradient problem and better capture long-term dependencies within sequential data.

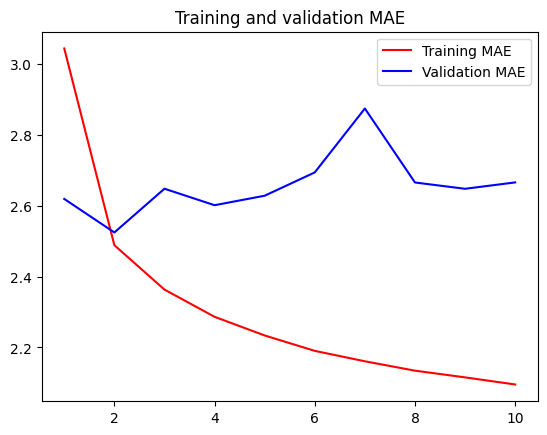
**Data Normalization:**

Here we would like to give a brief idea about Data preprocessing, it involves standardizing the data by subtracting the mean and dividing it by the standard deviation of the training set. After normalization, the dataset is divided into separate sets for training, validation, and testing.

**Results:**

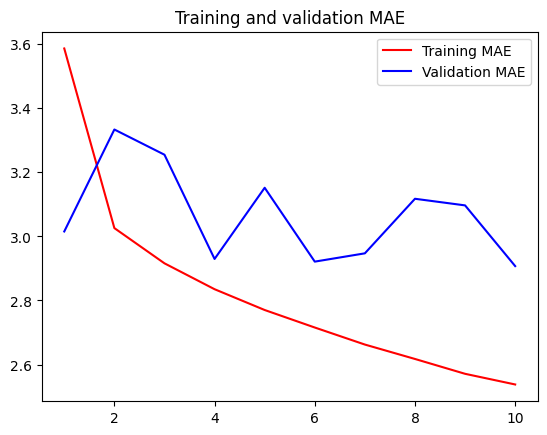
**Densely connected neural network:**

The **basic densely connected neural network model**, often referred to as a dense neural network, is a straightforward architecture where each neuron in one layer is connected to every neuron in the next layer. The model achieves a reported validation MAE of 2.66 and test MAE of 2.66 .



**1D Convolutional neural network:**

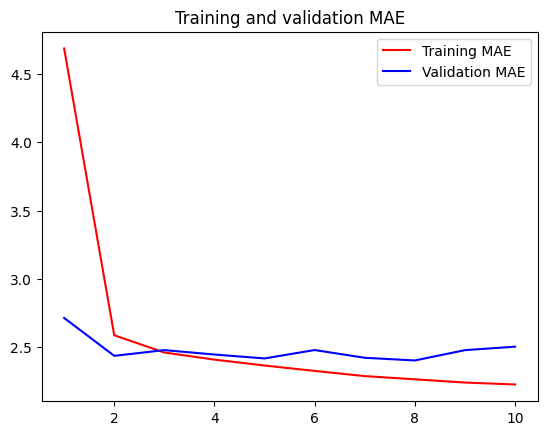
The **convolutional neural network (CNN) model**, specifically a 1D convolutional neural network, is designed to extract features from sequential data such as time series. In the context of time series forecasting with the Jena weather dataset, the 1D CNN model would employ one-dimensional convolutional layers to capture patterns and dependencies within the sequential data. The model achieves a reported validation MAE of 2.9 and test MAE of 3.16



**LSTM Models:**

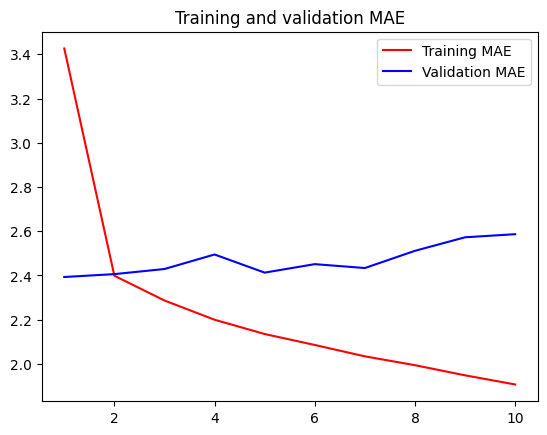
The **LSTM (Long Short-Term Memory) model**, a type of recurrent neural network (RNN), is well-suited for processing and predicting sequential data, making it a popular choice for time series forecasting tasks like the one using the Jena weather dataset.

**The Simple LSTM with Dense 16 model:**



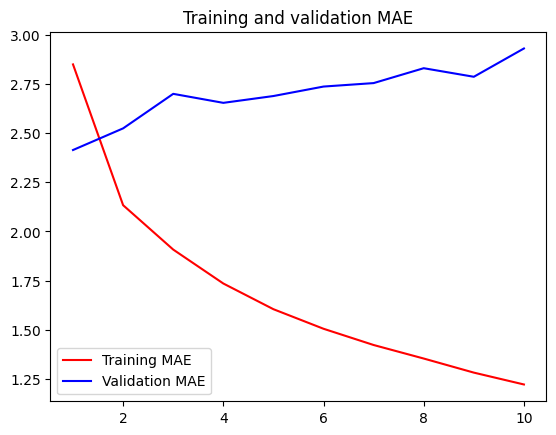
In this specific case, the reported validation MAE of 2.50 and test MAE of 2.52.

**The Simple LSTM with Dense 32 model:**



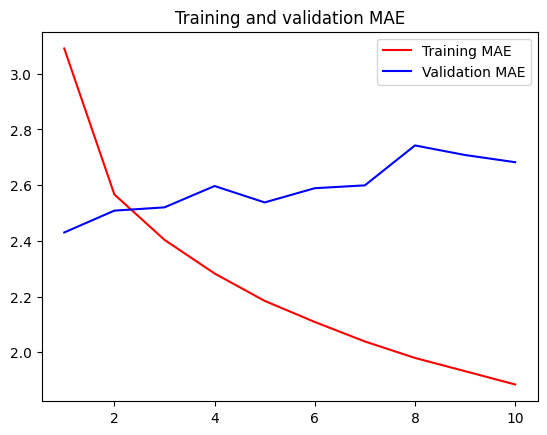
The reported performance metrics for are a validation MAE of 2.58 and a test MAE of 2.62.

**The Simple LSTM with Dense 64 model:**



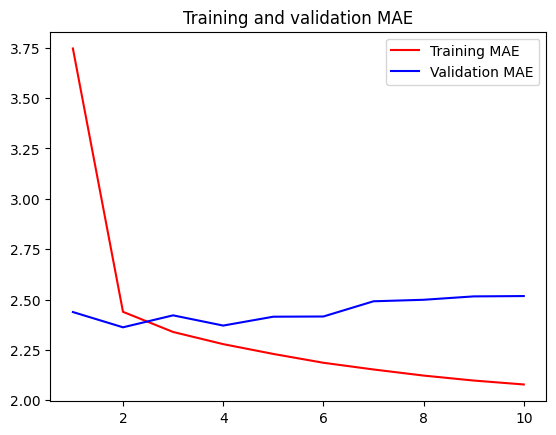
The reported performance metrics for the model are a validation MAE of 2.93 and a test MAE of 2.57.

**The Stacked LSTM with dropout model:**



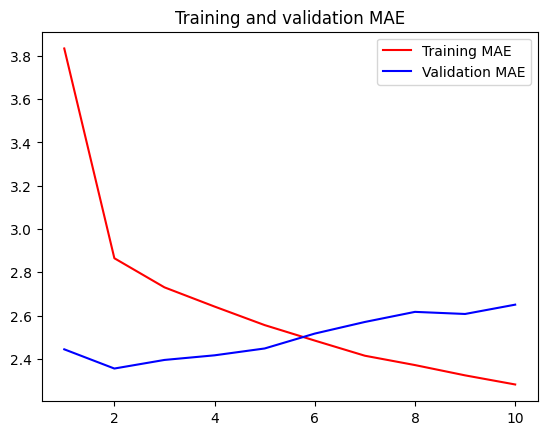
The reported performance metrics are a validation MAE of 2.65 and a test MAE of 2.59.

**The Bidirectional LSTM model:**



The reported performance metric for this model is a validation MAE of 2.51.

**1D Convnet and LSTM model:**



The reported performance metrics for the combination of this model are a validation MAE of 2.68 and a test MAE of 2.58.

Results:

|  |  |  |
| --- | --- | --- |
| **Model** | **Validation MAE** | **Test MAE** |
| Basic Dense | 2.68 | 2.69 |
| 1D Convolutional | 2.9 | 3.16 |
| Simple LSTM(Dense 16) | 2.5 | 2.52 |
| Simple LSTM(Dense 32) | 2.58 | 2.62 |
| Simple LSTM(Dense 64) | 2.93 | 2.57 |
| Stacked LSTM with Dropout | 2.65 | 2.59 |
| Bidirectional LSTM | 2.51 | - |
| Combination of 1D\_Convnet and LSTM | 2.68 | 2.58 |

Conclusion:

In this time series forecasting analysis using the Jena weather dataset, a variety of deep learning models were explored and compared. The results clearly demonstrate the superiority of recurrent neural network (RNN) architectures, particularly LSTMs and bidirectional LSTMs, over simpler models like dense neural networks and 1D convolutional networks.

The basic dense neural network and 1D CNN models struggled due to their inability to effectively capture the temporal dependencies present in the sequential data. By flattening or disrupting the time series information through operations like pooling, these models failed to leverage the sequential patterns crucial for accurate forecasting.

In contrast, the LSTM and bidirectional LSTM models, by design, could model long-range dependencies and maintain the temporal context through their recurrent connections and gating mechanisms. Techniques like recurrent dropout and bidirectional processing further enhanced the LSTM performance.

Among all the models evaluated, the bidirectional LSTM emerged as the top performer, achieving the lowest mean absolute error of 2.51 on the validation set. Its bidirectional nature allowed it to better learn patterns by processing the data in both forward and reverse directions.

While the GRU model demonstrated computational efficiency, fine-tuning the LSTM architectures by adjusting hyperparameters like the number of units and dropout rates proved more effective for improving forecasting accuracy on this time series task.

In summary, the analysis highlights the importance of leveraging advanced RNN capabilities like LSTMs and bidirectional processing to capture long-range temporal dependencies, a critical requirement for precise time series forecasting of sequential data such as weather data. The bidirectional LSTM model stood out as the most suitable architecture for this forecasting problem on the Jena weather dataset.

Top of Form

Top of Form