Time-Series Data

Name:AVS Praveen reddy

Student Id: 811254379

**Introduction:**

The aim of the project is to predict the temperature based on the use of time-series data and RNNs. Thus, such analysis is done between three neural network architectures: Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and a hybrid Conv1D + LSTM model. In the below code, each model has been tested for its ability to predict temperature values with high precision.

**Background:**

Time-series forecasting finds applications in domains like weather prediction, where the forecast of values in the future based on historical patterns becomes paramount. RNNs are especially appropriate in this regard, since they keep track of temporal information and thus can make informed predictions over time. This project hence identifies which model architecture performs the best for temperature prediction.

**Dataset and Preprocessing:**

The dataset, contains various weather-related features. Key preprocessing steps included:

1. **Feature Selection:** Selected features relevant to forecasting, such as Temperature, Humidity, Wind Speed, and Pressure.
2. **Handling Missing Values:** Filled missing values using forward filling to maintain data continuity.
3. **Normalization:** Scaled features to a 0-1 range using MinMaxScaler to enhance model performance by reducing feature variance.
4. **Sequence Generation:** Generated sequences of 60 time steps for model input, with each sequence predicting the subsequent temperature value.

**Methodology:**

Implemented three different architectures for models and compared them:

* **LSTM Model:** It contains two layers of LSTMs with dropout regularization to prevent overfitting.
* **GRU Model**: The configuration is like that of the LSTM model, only using GRU layers.
* **Conv1D + LSTM Model**: A Conv1D for feature extraction is followed by the dense layers that help LSTM capture the local pattern in the data.

**Hyperparameters and Training**

* **Epochs:** 25 epochs with early stopping to prevent overfitting.
* **Batch Size:** 32, balancing computational efficiency and memory usage.
* **Early Stopping:** Configured to halt training if the validation loss does not improve over 10 consecutive epochs.

**Evaluation Metrics:**

Each model’s performance was assessed using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), with lower values indicating better performance.

**Results:**

|  |  |  |
| --- | --- | --- |
| **Model** | **MAE** | **RMSE** |
| **LSTM** | 0.24712 | 0.35072 |
| **GRU** | 0.7312 | 0.9430 |
| **Conv1D + LSTM** | 1.737 | 2.1064 |

**Screenshots:**

**Sample Visualizations**

1. **LSTM Temperature Forecasting:** (Include LSTM forecast plot)
2. **GRU Temperature Forecasting:** (Include GRU forecast plot)
3. **Conv1D + LSTM Temperature Forecasting:** (Include Conv1D + LSTM forecast plot)

**1.**

**A graph showing the temperature of a weather forecast

Description automatically generated**

**2.**

**A graph showing the weather forecast

Description automatically generated**

**3.**

A graph showing the temperature of a person

Description automatically generated

**Conclusion:**

This project has shown that RNN architectures are suitable for time-series weather forecasting, notably LSTM and GRU. Each model has an advantage: the LSTM and GRU models were advantageous in sequence-based predictions, and the Conv1D + LSTM model could utilize the feature extraction capabilities. Among them, according to MAE, the best performing architecture on this dataset proved to be the LSTM model.