Time-Series Data Assignment: RNN Application for Weather Forecasting

Introduction

This report presents the results of applying Recurrent Neural Networks (RNNs) to weather time-series forecasting. The assignment focused on predicting temperature values 24 hours in advance using 5 days of historical weather data from the Jena Climate dataset (2009-2016). Various RNN architectures were systematically tested to improve the forecasting performance.

Data and Methodology

Dataset

The Jena Climate dataset contains 420,551 data points of weather measurements taken at 10-minute intervals. For this analysis:

- Data was sampled at hourly intervals (sampling rate = 6)
- Each input sequence used 120 time steps (equivalent to 5 days of data)
- The target was to predict temperature 24 hours into the future
- Data was split into training (50%), validation (25%), and test (25%) sets
- Features were normalized using mean and standard deviation from the training set

Baseline Method

To benchmark the performance of neural network models, a naive baseline was established:

- Prediction method: Last observed temperature value as the prediction for the future
- Validation MAE: 2.44
- Test MAE: 2.62

Model Architectures

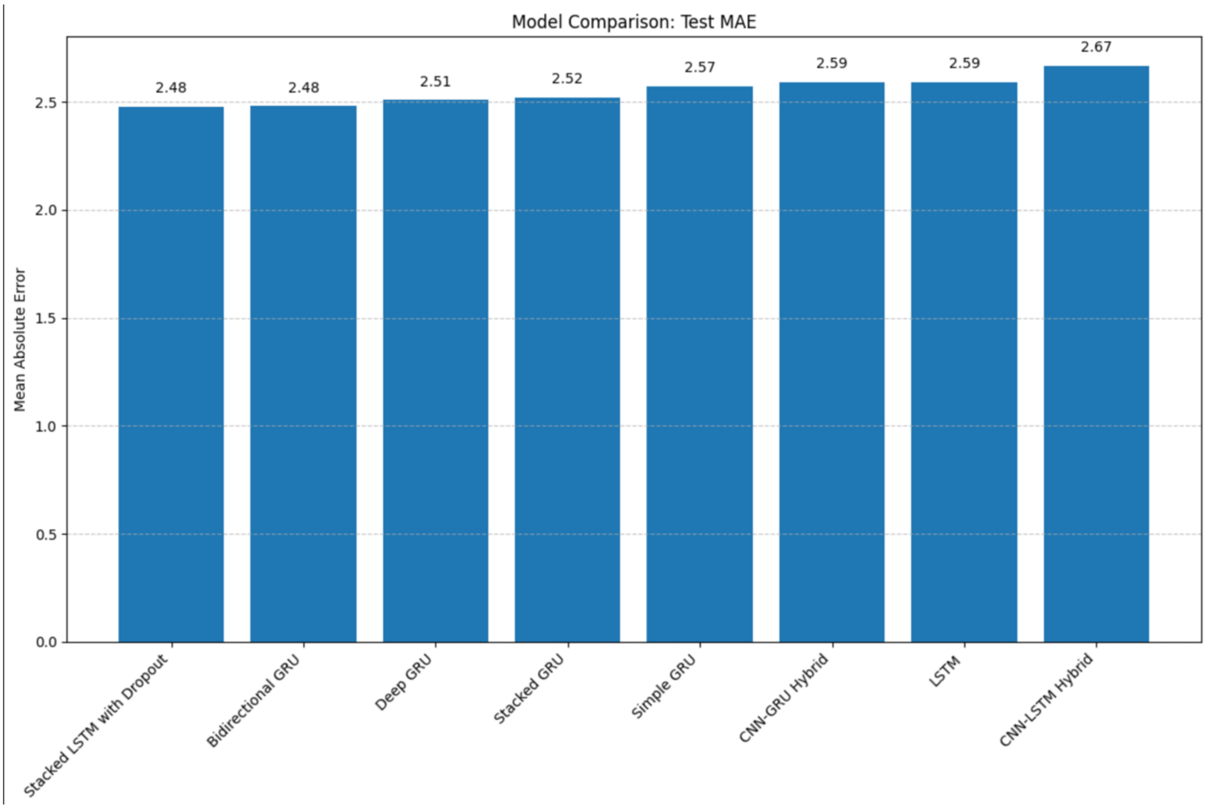
Multiple RNN architectures were implemented and evaluated:

1. **Simple GRU**: Single GRU layer with 32 units
2. **Stacked GRU**: Two GRU layers with 64 units each
3. **LSTM**: Single LSTM layer with 32 units
4. **Stacked LSTM with Dropout**: Two LSTM layers with 64 units and dropout rates of 0.3
5. **CNN-GRU Hybrid**: Convolutional layer followed by a GRU layer
6. **CNN-LSTM Hybrid**: Convolutional layer followed by an LSTM layer
7. **Bidirectional GRU**: Single bidirectional GRU layer with 32 units
8. **Deep GRU**: Three GRU layers with 128 units and dropout

All models were trained using:

- Adam optimizer
- Mean Squared Error (MSE) loss function
- Mean Absolute Error (MAE) as the evaluation metric
- Early stopping with patience=5 to prevent overfitting
- Model Checkpoint to save the best performing model

Results



Performance Comparison

Model Type	Test MAE
Stacked LSTM with Dropout	2.48
Bidirectional GRU	2.48
Deep GRU	2.51
Stacked GRU	2.52
Simple GRU	2.57
CNN-GRU Hybrid	2.59
LSTM	2.59
CNN-LSTM Hybrid	2.67

Key Findings

1. **Best Models**: Both the Stacked LSTM with Dropout and the Bidirectional GRU achieved the best performance with a test MAE of 2.48.
2. **Improvement over Baseline**: The best models showed a 5.39% improvement over the naive baseline approach (reducing MAE from 2.62 to 2.48).

3. **Model Complexity**: Increasing model complexity beyond a certain point showed diminishing returns. The Deep GRU model (3 layers, 128 units) performed slightly worse than simpler architectures with more appropriate regularization.
4. **Regularization**: The addition of dropout proved beneficial in the LSTM model, helping it achieve top performance.
5. **Bidirectional Processing**: Processing sequences in both directions (forward and backward) significantly improved predictions, suggesting temporal dependencies exist in both directions for temperature forecasting.
6. **CNN Preprocessing**: The addition of convolutional layers did not improve performance, indicating that local feature extraction is not particularly beneficial for this dataset. In fact, CNN-based hybrid models showed the worst performance overall.
7. **GRU vs LSTM**: Simple GRU and LSTM layers performed similarly, but when combined with additional techniques (stacking, dropout, bidirectional), they showed different response patterns.

Discussion

The results demonstrate that advanced RNN architectures can improve temperature forecasting compared to naive approaches, but the improvement margin is modest (5.39%). This suggests that:

1. Temperature prediction is inherently challenging with this approach, possibly due to chaotic nature of weather systems
2. There may be fundamental limitations to how accurately we can predict temperature 24 hours ahead using only historical temperature data
3. Further feature engineering or incorporating additional weather variables might be necessary for significant improvements
4. The relatively similar performance across different architectures suggests we might be approaching the limit of what can be achieved with this data and prediction horizon

The most effective approaches proved to be:

- Adding bidirectional capability to capture dependencies regardless of sequence position
- Implementing proper regularization through dropout to prevent overfitting
- Using stacked architectures with appropriate capacity (neither too simple nor too complex)

Conclusion

This analysis demonstrates that well-designed RNN architectures can modestly outperform simple baselines in temperature forecasting tasks. The Stacked LSTM with Dropout and Bidirectional GRU models achieved the best performance with a test MAE of 2.48, representing a 5.39% improvement over the baseline.

While the improvement might seem modest, it highlights the challenges inherent in time-series forecasting for complex natural phenomena like weather. The results also suggest that model architecture selection should be guided by:

1. Appropriate regularization strategies

2. Consideration of bidirectional dependencies in the data
3. Balancing model capacity with the risk of overfitting

Future work could explore longer sequence inputs, different prediction horizons, additional weather variables, or alternative deep learning architectures like Transformer models that have shown promising results in sequence modeling tasks.