Assignment 3: Time-Series Data
Group 24

This report presents the application of Recurrent Neural Networks (RNNs) to time-series data, specifically focusing on weather forecasting problems. The assignment aims to explore various methods to improve the performance of RNN models in forecasting weather patterns. The methods include adjusting the architecture of the RNN model, experimenting with different types of recurrent layers (e.g., LSTM, GRU), and incorporating 1D convolutions alongside RNN layers. The report outlines the implementation of these methods, evaluates their performance on validation datasets, and presents the best-performing models tested on the test set. Overall, the report aims to showcase the effectiveness of RNNs in handling time-series data and highlight strategies for improving their forecasting accuracy.

Summary of the models for time-series data

Model	Dense Units	Dropout	Loss	Test MAE
Basic Machine Learning model	16	No	11.6759	2.69
Basic Machine Learning model	64	No	11.8205	2.7
1D Convolution model	16	No	15.2386	3.07
RNN models				
Simple RNN	16	No	151.3941	9.93
Stacked Simple RNN Model	16	No	151.1465	9.91
GRU				
Simple GRU (Gated Recurrent Unit)	16	No	9.8051	2.44
LSTM(Long Short-Term Memory)				
LSTM-Simple	16	No	10.5209	2.53
LSTM - dropout Regularization	16	Yes	11.063	2.62
LSTM - Stacked setup with 16 units	16	No	11.0013	2.59
LSTM - Stacked setup with 32 units	32	No	11.388	2.64
LSTM - Stacked setup with 8 units	8	No	10.2133	2.51
LSTM - dropout-regularized, stacked model	8	Yes	11.1213	2.59
Bidirectional LSTM	16	No	10.8084	2.59
Combinations				
1D Convnets and LSTM together	16	No	23.255	3.83

Applying RNNs to Time-Series Data

- From the results, it's evident that simple RNN and stacked SimpleRNN models perform poorly in terms of MAE on the test set, with MAE values significantly higher than other models. This suggests that simple RNNs might not be suitable for this particular time-series forecasting task.
- GRU and LSTM models, on the other hand, perform better. Among them, the Simple GRU and Bidirectional LSTM models have the lowest test MAE values, indicating their effectiveness in capturing the temporal patterns in the data.
- LSTM models with different configurations (dropout regularization, stacked setup with varying units) also show reasonable performance, although they are not the top performers.

Improving Performance of the Network for Time-Series Data

- Dropout regularization is applied in some LSTM models, aiming to prevent overfitting. However, the LSTM models with dropout regularization don't necessarily outperform those without dropout.
- Increasing the number of units in the LSTM stacked setup doesn't consistently lead to better performance. For instance, the LSTM stacked setup with 32 units has a slightly higher test MAE compared to the setup with 16 units.
- Bidirectional LSTM, which captures information from both past and future time steps, shows promising results compared to unidirectional LSTM models.

Applying Different Deep Learning Layers to Time-Series Data

The combination of 1D Convolutional Neural Networks (CNNs) and LSTM shows worse performance compared to individual LSTM or GRU models. This suggests that for this specific task, CNNs might not be as effective in capturing relevant features from the time-series data.

Recommendations

- Focus on models that perform well on the test set, such as Simple GRU and Bidirectional LSTM.
- Experiment with different architectures and hyperparameters to find the best-performing model for this specific dataset.
- Consider incorporating additional features or engineering new features that might improve the model's ability to capture relevant patterns in the data.
- Regularly evaluate models on both validation and test sets to ensure that performance improvements generalize well beyond the training data.
- Explore other deep learning techniques tailored for time-series forecasting, such as attention mechanisms or hybrid models combining traditional statistical methods with deep learning approaches.