README

Recurrent Neural Networks, Text Generation, and Climate Analysis

Author: Shaurya Handu
Tools: Python, NumPy, Pandas, TensorFlow/Keras, Matplotlib, Scikit-learn
Environment: Jupyter Notebook / Kaggle

1. Project Overview

This project presents a unified study of **sequence modeling using Recurrent Neural Networks (RNNs)** and their gated variants, namely **GRU** and **LSTM**. The models are applied across two domains:

- Character-level text generation using Office dialogue data
- Climate time-series analysis and forecasting for Manipal

The central objective is to understand temporal dependency learning and evaluate model behavior across domains.

2. Theoretical Foundation

The project builds upon a formal understanding of recurrent architectures:

- **Vanilla RNN**: Sequential hidden-state propagation with shared parameters
- **Vanishing Gradient Problem**: Difficulty in learning long-term dependencies
- **GRU**: Update and reset gates to regulate information flow
- **LSTM**: Dedicated cell state with forget, input, and output gates

These concepts are mathematically derived and discussed in the accompanying technical report.

3. Text Generation Project

3.1 Objective

To build a character-level language model capable of generating coherent dialogue resembling Office-style scripts.

3.2 Methodology

- Character-level tokenization
- Vocabulary construction from unique characters
- Comparison of RNN, GRU, and LSTM architectures

3.3 Training Enhancements

- Increased number of epochs
- Early stopping based on validation loss
- Visualization of training and validation loss curves

3.4 Outcome

The LSTM model achieved the lowest perplexity and produced the most coherent long-range text.

4. Climate Time-Series Analysis

4.1 Objective

To analyze long-term temperature and precipitation trends in Manipal using an LSTM-based forecasting model.

4.2 Data Processing

- Date-time indexing
- Min-Max normalization
- Sliding window sequence generation (12-month lookback)

4.3 Model Architecture

- Single-layer LSTM with 64 hidden units
- Tanh activation function
- Adam optimizer with MSE loss

4.4 Results

The model successfully captured:

- Strong annual rainfall seasonality
- A positive long-term temperature trend
- Increasing precipitation variability

5. Cross-Domain Insight

A key contribution of this work is the reuse of the same LSTM architecture for:

- Natural language modeling
- Climate forecasting

This demonstrates the generality and robustness of LSTMs for sequential data.

6. Visualizations

Model performance and data behavior were analyzed using:

- Training vs validation loss curves
- Seasonal climate plots
- Prediction vs ground-truth comparisons

7. Conclusion

This project provides a complete theoretical-to-practical pipeline for recurrent neural networks, highlighting their strengths, limitations, and versatility across domains involving temporal data.

This README summarizes cumulative experimentation, iteration, and analysis conducted across multiple notebooks and a formal technical report.