**CS583 Homework Assignment: Exploring and Optimizing RNNs**

**Objective:**

Implement, extend, and analyze a Recurrent Neural Network (RNN) to predict sequences. Reflect on your work through a detailed report.

**Tasks:**

1. **Data Preparation**:
   * Load the datasets:
     + Reverse\_Arithmetic\_Sequences.csv
     + Key\_Sequences.csv
     + The python code for generating the datasets is also attached.
   * Preprocess the datasets:
     + Normalize or scale **Reverse Arithmetic Sequences**.
     + One-hot encode **Key Sequences**.
2. **RNN Implementation**:
   * Modify the provided RNN code to:
     + Train and test the model on **Reverse Arithmetic Sequences**.
     + Train and test the model on **Key Sequences**.
3. **Hyperparameter Exploration**:
   * Experiment with hyperparameters such as:
     + Hidden layer size.
     + Number of layers.
     + Learning rate.
     + Sequence length (input size).
   * Analyze the impact of these changes on the model’s performance.
4. **Error Analysis**:
   * Evaluate model performance on a test set. For instance, you can divide the datasets into 70% for training, 10% for validation, and 20% for testing.
   * Analyze and describe:
     + Which types of sequences the model performs poorly on.
     + Potential reasons for these errors.
5. **Write a Detailed Report**:
   * **Implementation Details**: Discuss the adaptations made to the RNN code and the rationale behind your design choices.
   * **Hyperparameter Tuning**: Summarize your hyperparameter experiments and observations.
   * **Challenges and Solutions**: Highlight any challenges and your approach to solving them.
   * **Error Analysis**: Describe patterns in errors and suggest improvements to the model.
   * **Lessons Learned**: Reflect on your understanding of RNNs and how this assignment contributed to it.

**Deliverables:**

* **Code**: A Jupyter Notebook file containing:
  + Data preprocessing steps.
  + Adapted RNN code with hyperparameter tuning.
  + Visualizations of training loss and evaluation results.
* **Report**: A detailed (2 page) document including:
  + Implementation details and design choices.
  + Results of hyperparameter tuning.
  + Error analysis and suggestions for improvement.
  + Key takeaways and lessons learned.

**Grading Criteria:**

* Data Preprocessing: 20%
* RNN Implementation: 30%
* Hyperparameter Exploration: 20%
* Visualization and Evaluation: 15%
* Report: 15%