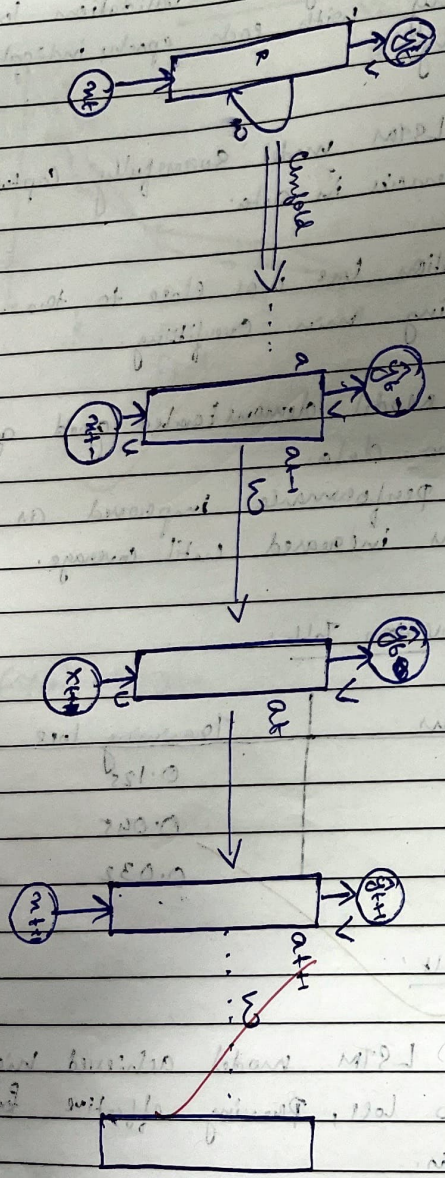
RNN Architecture



LAB-9 Build a Recurrent Neural Network

Aim:

To design and train a Recurrent Neural Network (RNN) for sequence data prediction.

Objectives:

1. Prepare and preprocess sequential dataset.
2. Build and train an RNN model.
3. Evaluate model performance using loss and accuracy metrics.

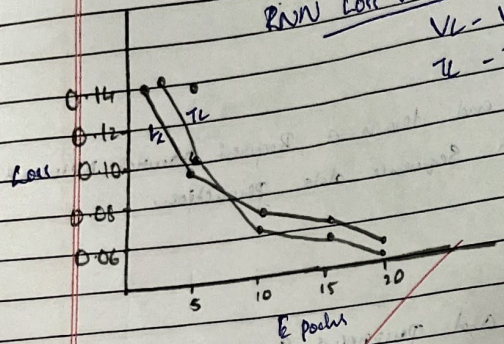
Pseudo Code:

- Load dataset.
- preprocess → tokenize/normalize sequences.
- Split into training and testing sets.
- Initialize RNN Model (Simple RNN layer)
- Compile (Optimizer = "adam", loss = "mse")
- Train model with epochs.
- Evaluate performance metrics.
- Save trained RNN model.

RNN Loss Vs Epochs

VL - Validation Loss

TL - Training Loss



Observation:

1. The RNN Model showed a gradual decrease in loss over epochs.
2. The network efficiently learned temporal patterns in the data.
3. The validation loss remained close to training loss, indicating stability.

Observation Table:

Epochs	Training loss	Validation loss
1	0.142	0.138
10	0.065	0.069
20	0.048	0.052

Result:

The RNN Model effectively captured sequential relationships and produced accurate predictions