Recurrent Neural Networks (RNNs) - A Brief Overview

What is an RNN?
A Recurrent Neural Network (RNN) is a type of neural network designed for processing sequential data, such as ti

Key Concepts:
- Hidden State: Each RNN cell maintains a hidden state vector h(t) that captures information from previous steps.
- Sequence Processing: Inputs are processed one at a time (e.g., word by word in a sentence), and the network's c
- Parameter Sharing: RNNs use the same weights at each time step, reducing the number of parameters.

Mathematics Behind RNNs:
For input x(t) and hidden state h(t) at time step t:

$h(t) = \tanh(W_{xh} * x(t) + W_{hh} * h(t-1) + b_h)$
$y(t) = W_{hy} * h(t) + b_y$

Where:
- W_{xh}, W_{hh}, W_{hy} are weight matrices
- b_h, b_y are bias vectors
- tanh is an activation function

Problems with Vanilla RNNs:
- Vanishing and Exploding Gradients: Hard to capture long-term dependencies.
- Short Memory: Struggles to remember information from far back in the sequence.

Variants of RNNs:
- LSTM (Long Short-Term Memory): Uses gates to control information flow.
- GRU (Gated Recurrent Unit): A simplified, efficient version of LSTM.

Applications:
- Natural Language Processing (translation, sentiment analysis)
- Time Series Forecasting
- Speech Recognition
- Music Generation