RNNs Summary

RNN stands for **Recurrent Neural Network**, and it's a type of neural network that's made for working with **sequential data**, like text, time series, or anything where the **order matters**.

The key idea behind RNNs is that they don't just take one input and forget about it, they remember things from the past using something called a **hidden state**. This hidden state gets updated at each time step and helps the network keep track of what came before. So, it's kind of like the model has a memory of previous inputs.

At each time step, the RNN takes the current input and the previous hidden state, and it outputs a new hidden state (and sometimes an output). The same set of weights is reused across all the time steps, which makes it different from regular neural networks.

I also searched about if RNNs can have more than one layer, when they do, it's called a **stacked RNN**. This makes the model deeper and helps it learn more complex patterns in the sequence.

One issue with RNNs is that they struggle with **long-term dependencies** — they’re not great at remembering things from way back in the sequence. That’s why there are improved versions like **LSTMs and GRUs**, which are better at holding on to long-term information.