# 🌀 Types of RNNs and Bidirectional RNNs

Recurrent Neural Networks (RNNs) are sequence models designed for data where **order matters** (time series, text, audio, video).

Depending on **input-output mapping**, RNNs can be classified into 4 main types:

## 1. **One-to-One RNN**

* Standard **feed-forward neural network**.
* One input → one output.
* Used in **classification tasks** (e.g., image → label).

**Example**: Handwritten digit classification (MNIST).  
Input: Image (28x28).  
Output: Digit label (0–9).

## 2. **One-to-Many RNN**

* One input → sequence of outputs.
* The input vector is used to generate multiple outputs over time.

**Example**:  
- Music generation (input = style, output = sequence of notes).  
- Image captioning (input = image features, output = words in a sentence).

## 3. **Many-to-One RNN**

* Sequence of inputs → one output.
* The RNN processes the sequence and summarizes it in a single prediction.

**Example**:  
- Sentiment analysis (input = sentence, output = sentiment [positive/negative]).  
- Fraud detection (input = sequence of transactions, output = fraud/not fraud).

## 4. **Many-to-Many RNN**

* Sequence of inputs → sequence of outputs.
* Two sub-types:
  1. **Equal length input/output** → e.g., Part-of-speech tagging (word → POS tag).
  2. **Different length input/output** → e.g., Machine translation (English sentence → French sentence).

# 🔀 **Bidirectional RNN (BiRNN)**

A **vanilla RNN** processes the sequence only in **one direction** (past → future).  
But sometimes, the **future context** is also useful.

👉 Example:  
Sentence = “He said **bank** …”  
- If only reading left → right, “bank” might mean “river bank”.  
- If reading right → left as well, the next words (“to deposit money”) reveal that it’s a financial bank.

This is why we use **Bidirectional RNNs**.

## 📌 Concept of BiRNN

* BiRNN = Two RNNs:
  + One processes **forward (→)**
  + One processes **backward (←)**
* Their outputs are **merged** (concatenation, sum, or average).

So the hidden state at time step t is: [ h\_t = [h\_t^{→}; h\_t^{←}] ]

## 🔹 Forward Propagation Equations

For a sequence (x = (x\_1, x\_2, , x\_T)):

### 1. Forward RNN (→):

[ h\_t^{→} = f(W\_{xh}^{→} x\_t + W\_{hh}^{→} h\_{t-1}^{→} + b\_h^{→}) ]

### 2. Backward RNN (←):

[ h\_t^{←} = f(W\_{xh}^{←} x\_t + W\_{hh}^{←} h\_{t+1}^{←} + b\_h^{←}) ]

### 3. Merge:

[ h\_t = [h\_t^{→}; h\_t^{←}] ]

### 4. Output:

[ y\_t = g(W\_{hy} h\_t + b\_y) ]

Where: - (f) = hidden activation (tanh, ReLU)  
- (g) = output activation (softmax for classification, sigmoid for binary, etc.)

# 📝 Example of BiRNN

**Task**: Named Entity Recognition (NER)  
Sentence: “Steve Jobs founded Apple.”

* Input: sequence of words [Steve, Jobs, founded, Apple]
* Output: entity tags [PER, PER, O, ORG]

👉 Using BiRNN: - “Steve” → future context helps (“Jobs” is surname).  
- “Apple” → future context is missing in left-to-right, but backward pass confirms it’s a company.

Thus BiRNN improves accuracy.

# 🔒 Important Notes about BiRNNs

✅ **Advantages**: - Captures both **past & future context**.  
- More accurate in NLP tasks like **NER, POS tagging, speech recognition, translation**.

❌ **Limitations**: - Requires the **entire sequence** (not suitable for real-time streaming where future input is unknown).  
- Computationally heavier (two RNNs).

# 🔹 Extension: BiLSTM and BiGRU

BiRNNs can be extended with: - **BiLSTM** (Bidirectional LSTM): adds **gates (forget, input, output)** → solves vanishing gradient problem.  
- **BiGRU** (Bidirectional GRU): lighter version of LSTM, fewer gates.

Example equation for BiLSTM (forward cell): [ f\_t = (W\_f [h\_{t-1}, x\_t] + b\_f) ] [ i\_t = (W\_i [h\_{t-1}, x\_t] + b\_i), ; = (W\_C [h\_{t-1}, x\_t] + b\_C) ] [ C\_t = f\_t C\_{t-1} + i\_t ] [ o\_t = (W\_o [h\_{t-1}, x\_t] + b\_o), ; h\_t = o\_t (C\_t) ]

👉 In BiLSTM, both forward and backward versions exist, and their outputs are concatenated.

# 🏗️ Applications of RNN + BiRNN

* **One-to-One**: Image classification
* **One-to-Many**: Music generation, image captioning
* **Many-to-One**: Sentiment analysis, fraud detection
* **Many-to-Many**: Machine translation, speech-to-text
* **Bidirectional RNNs**: NER, POS tagging, speech recognition, translation

✅ This document provides a **full roadmap of RNN types + Bidirectional RNN (equations + examples + extensions to LSTM/GRU)**.