Feature	Feedforward Neural Network (FNN)	Recurrent Neural Network (RNN)
Architecture	Information flows in one direction (input to output)	Information flows in loops, with connections between time steps
Memory	No memory of past inputs; each input is processed independently	Maintains a hidden state that acts as memory of past inputs
Data Type	Suitable for static data (e.g., images, tabular data)	Suitable for sequential data (e.g., time series, language)
Input-Output Dependency	Input and output are independent of sequence	Output at each step depends on the previous inputs and states
Connection Between Layers	Fully connected layers without feedback loops	Includes feedback loops, allowing information to persist across time steps
Training	Uses standard backpropagation	Uses Backpropagation Through Time (BPTT) due to sequence dependency
Handling of Long-Term Dependencies	Limited in capturing temporal dependencies	Can capture temporal dependencies, though limited by vanishing gradients
Common Applications	Image recognition, object detection, tabular data analysis	Speech recognition, language modeling, machine translation
Vanishing/Exploding Gradient Problem	Less pronounced, due to fewer layers in depth	Common, especially with longer sequences, due to repeated backpropagation
Computational Complexity	Typically faster, no sequence dependency	Slower, due to sequential processing of data
Advanced Variants	Convolutional Neural Networks (CNNs), Multilayer Perceptron (MLP)	Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU)
Temporal Pattern Learning	Cannot capture temporal patterns effectively	Designed to capture temporal and sequential patterns
Output Dependence	Output is generated based solely on the current input	Output depends on the current input and past inputs