

Bi-directional RNN:

Process data in both forward and backward directions.

Capture contextual dependencies in the input data by considering past and future context.

Two separate RNNs that process data in opposite direction, output of these RNNs are combined to produce the final output.

Predict the next word .

Apple is my favorite _____ , and I work here

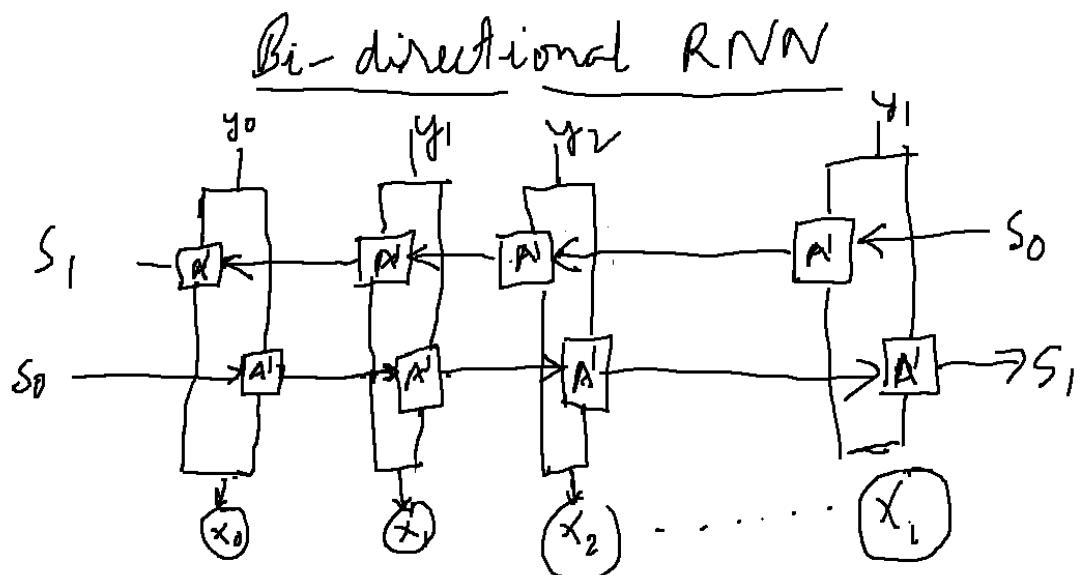
Apple is my favorite _____ , I am going to buy one

Apple could be fruit , company or phone.

Which bank?

Fisherman went to bank to deposit money.

Fisherman went to bank for fishing.



Back propagation through time(BPTT) : Algorithm specially designed to handle the temporal nature of RNNs.

During the forward pass, the forward RNN processes the input sequence in the usual way and it makes prediction. This prediction is compared with actual sequence and the error is back-propagated to update weights and bias

During backward pass RNN process the input in the reverse order and predict the output sequence. These predictions are compared with the actual sequence in reverse order. Error is back propagated through the network to update the weights and bias.

Once both passes are complete , the weights and bias are updated based on the results of forward and backward passes. This process is repeated until the model converges and predictions are same as values.

The model learns from past and future time steps when making predictions. This improves the model's accuracy.

Difference between RNN and Bi-directional RNN(BRNN)

RNN	BRNN
Process input in one direction	Process input in both forward and backward direction
Output at each time step depends only on the past inputs	Output at each time step depends on past and future inputs.
Trained on a single sequence	Trained on both forward and backward sequence.

Application Of BRNN:

Natural language processing : Language translation , Sentiment Analysis (twitter message whether it is positive , negative , neutral).

Time series forecasting such as predicting stock prices or weather patterns.

Advantages of BRNN:

Improved performance because they consider information from past and future time.

Ability to capture long term dependencies

Better handling of complex input sequences.

Disadvantages:

Increased computational complexity because there are two separate RNNs.

BRNN require more memory.

Training BRNN can be more difficult.

References:

<https://blog.paperspace.com/bidirectional-rnn-keras/>