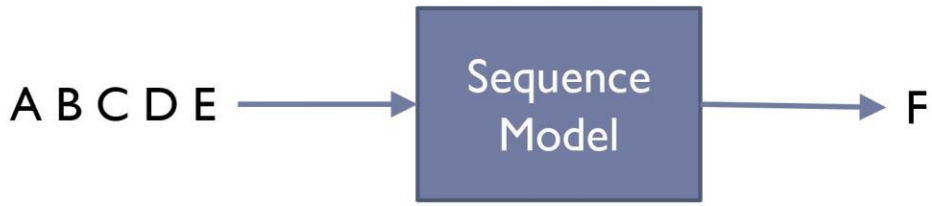
Sequence Modeling

Sequnce 2 Sequence Model

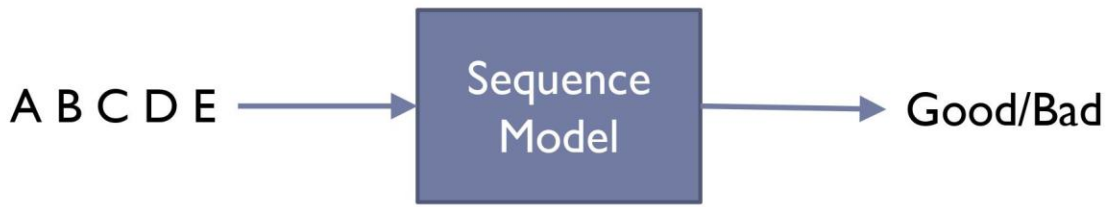
성균관대학교 소프트웨어학과
이 지 형

Sequential Data Modeling

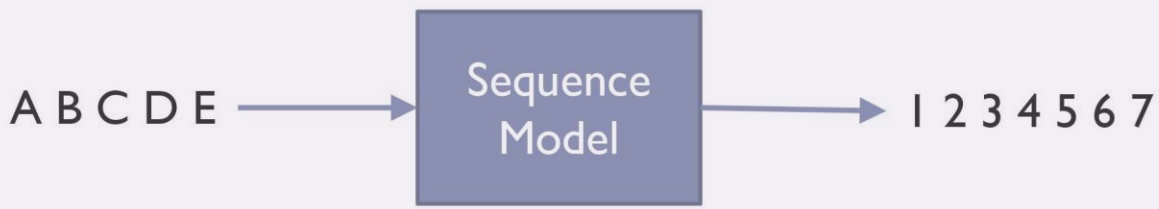
- Three Types of Problems
  - Next Step Prediction



  - Classification



  - Sequence Generation

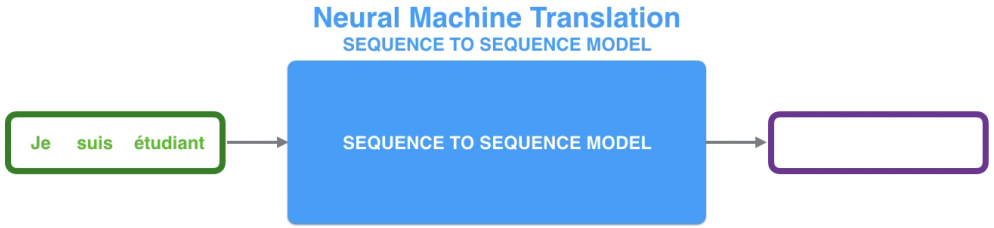


Sequence to Sequence Learning

Alamar (Attention)

- Sequence-to-sequence model (Sutskever et al., 2014, Cho et al., 2014)
  - ✓ A model that takes a sequence of items (words, letters, features of an images, etc.)
  - ✓ Outputs another sequence of items
    - A trained model

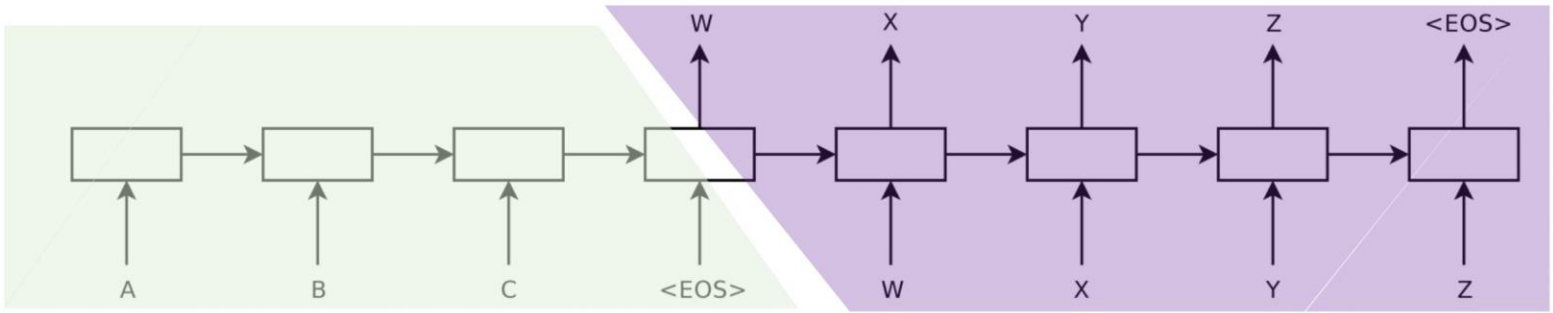
    - Neural machine translation



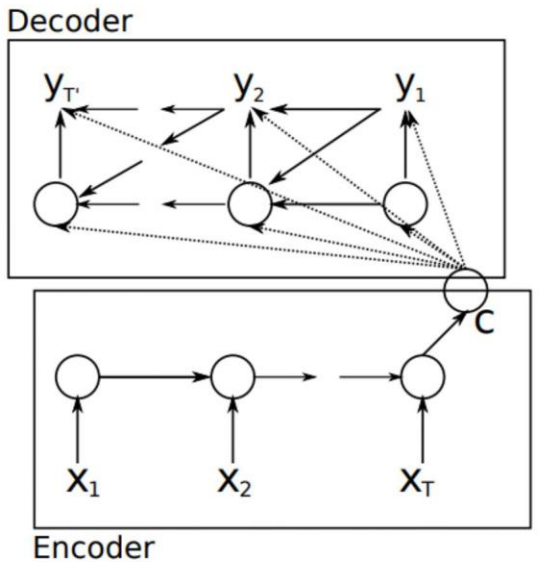
이 자료는 고려대학교 강필성 교수님의 “비정형 데이터 분석” 강의자료를 가져왔음을 밝힙니다.

Sequence to Sequence Learning

- Main idea
  - ✓ Seq2Seq model consists of an encoder and a decoder



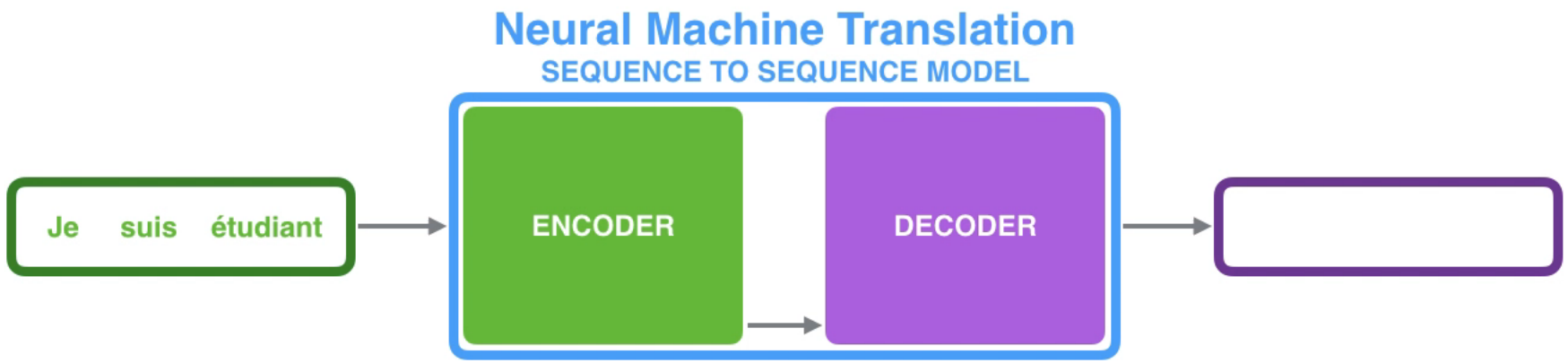
Sutskever et al., 2014



Cho et al., 2014

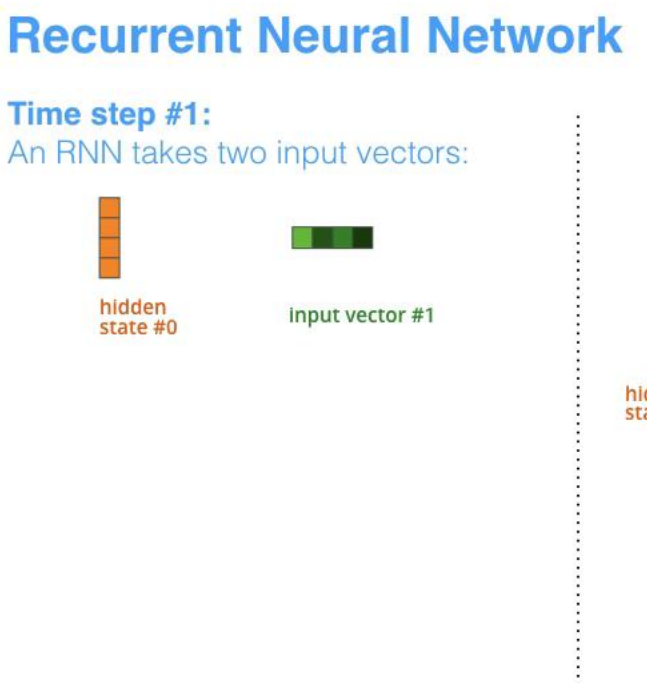
Sequence to Sequence Learning

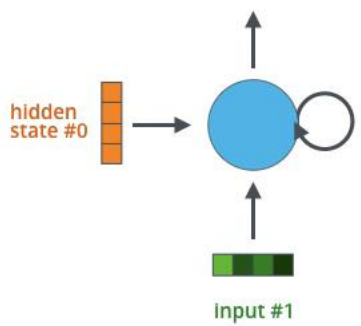
- Encoder-Decoder
  - ✓ The encoder processes each item in the input sequence and compiles the information it captures into a vector (context)
  - ✓ After processing the entire input sequence, the encoder send the context over to the decoder, which begins producing the output sequence item by item

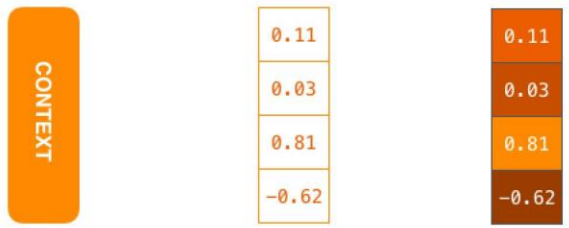


Sequence to Sequence Learning

- Encoder-Decoder
  - ✓ Recurrent neural network (RNN) is commonly used for the encoder and decoder structure
  - ✓ The context is a vector in the case of machine translation



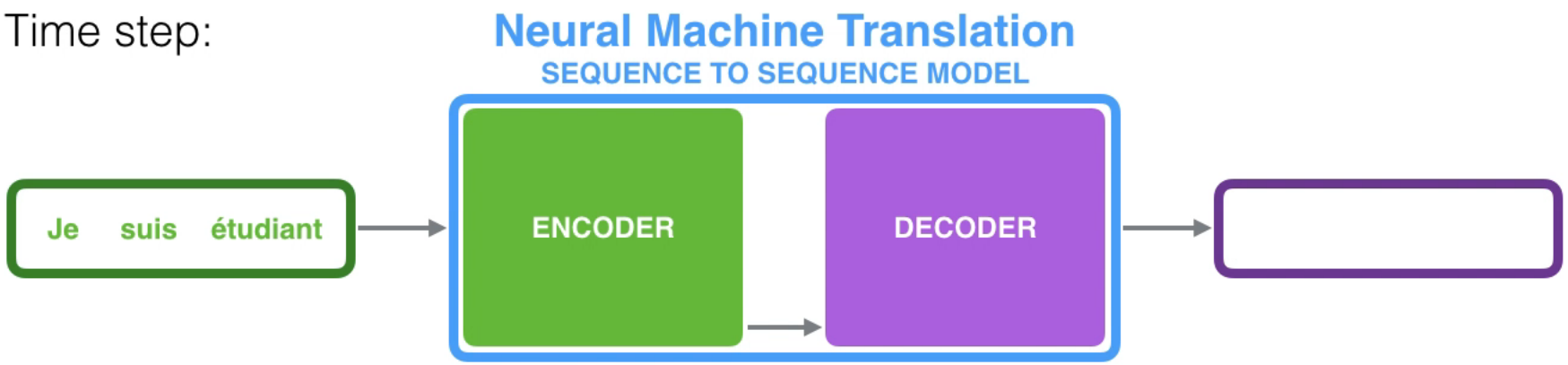




Sequence to Sequence Learning

Alamar (Attention)

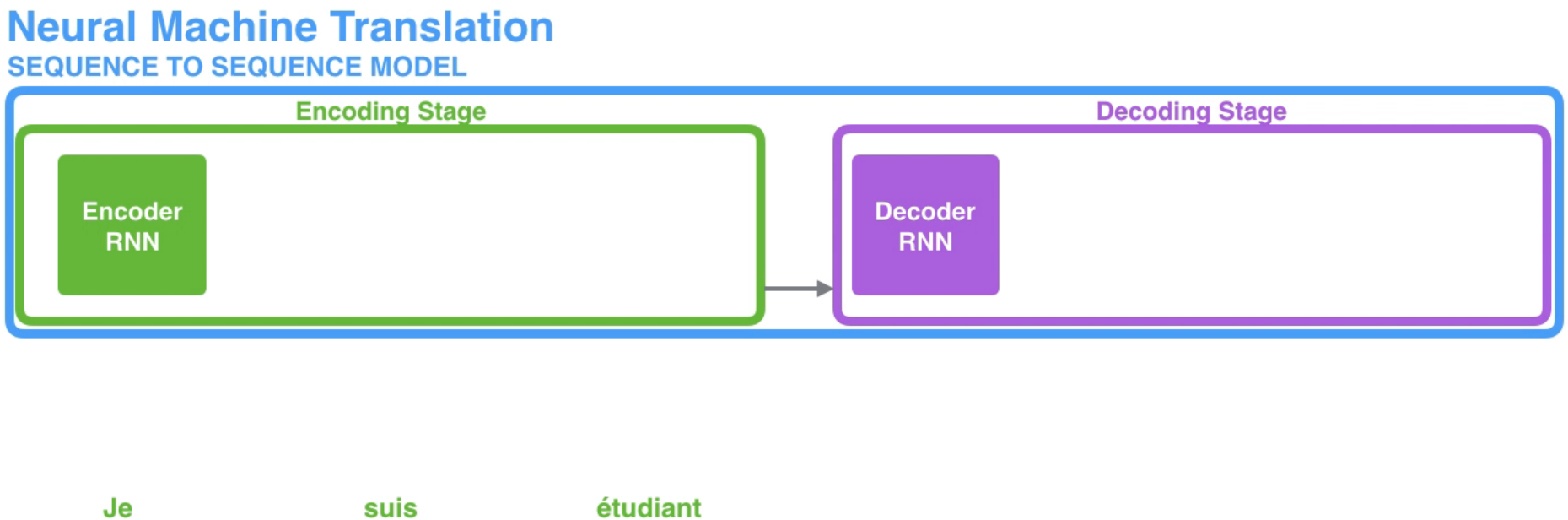
- Encoder-Decoder
  - ✓ Each purse for the encoder or decoder is that RNN processes its inputs and generates an output for that time step



Sequence to Sequence Learning

Alamar (Attention)

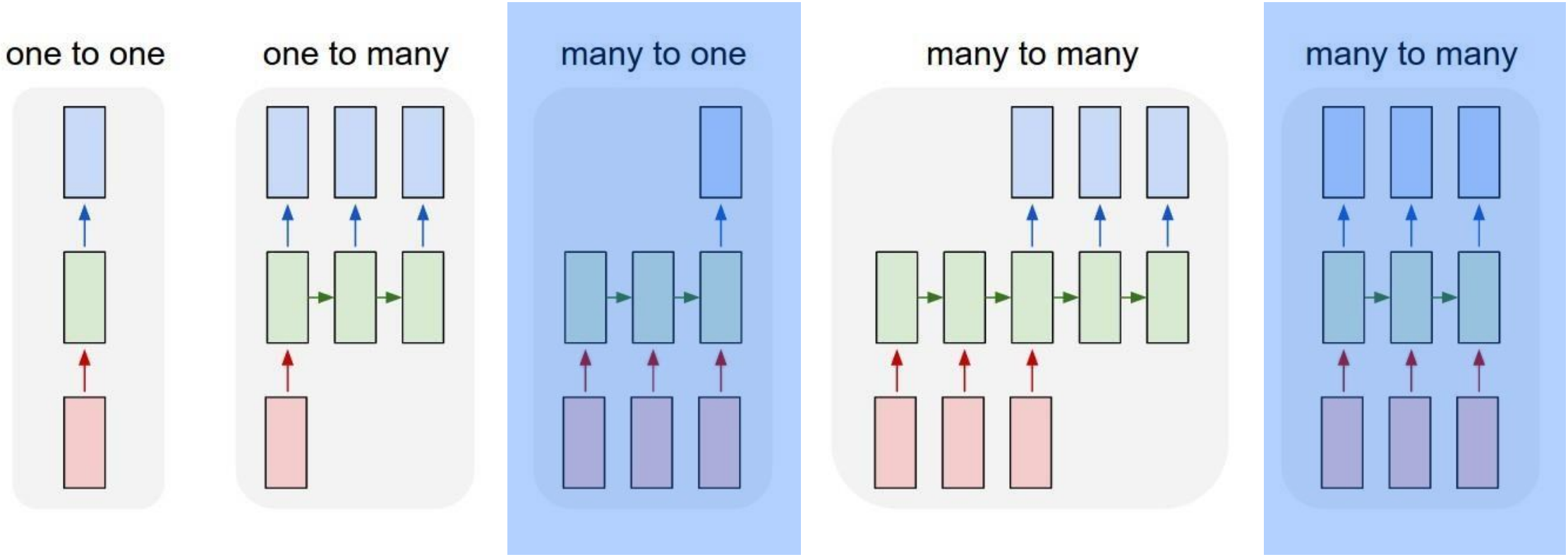
- An unrolled view of Seq2Seq learning



Deep Learning Dealing with Sequential Data

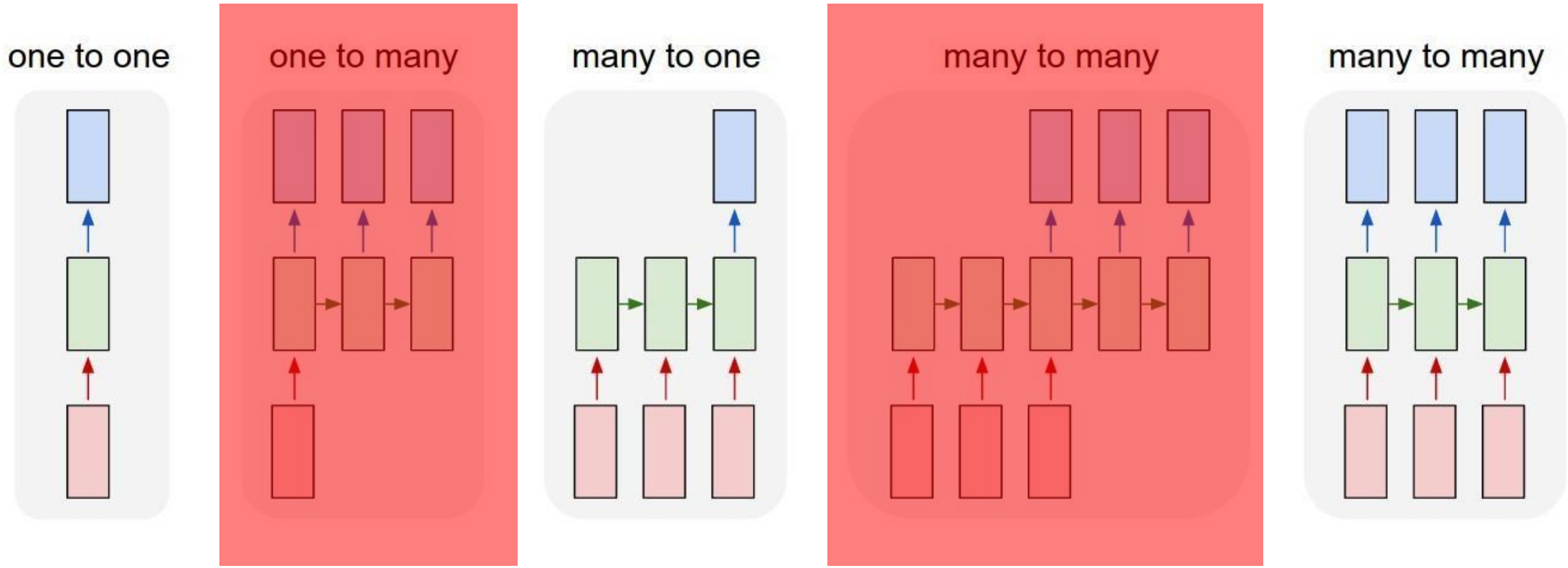
이 자료는 2019 KAIST idea factory 의 " 딥러닝 홀로서기” 자료를 대부분 참고하였음을 밝힙니다.

Types of Task Dealing with Sequential Data



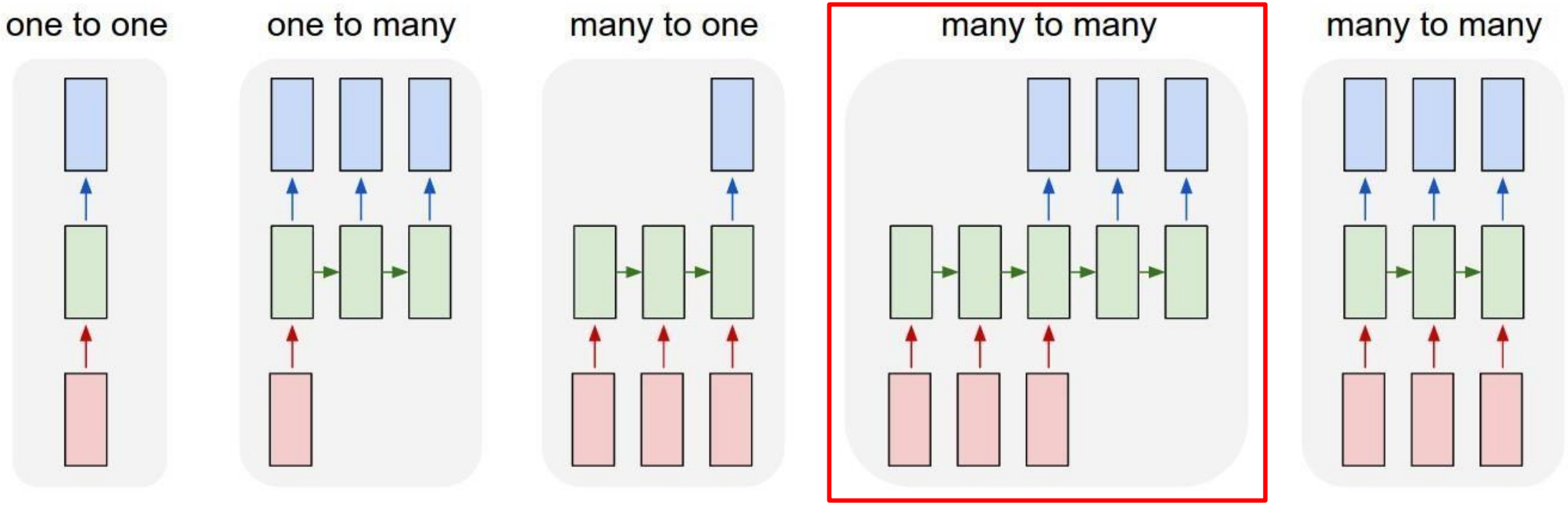
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Types of Task Dealing with Sequential Data



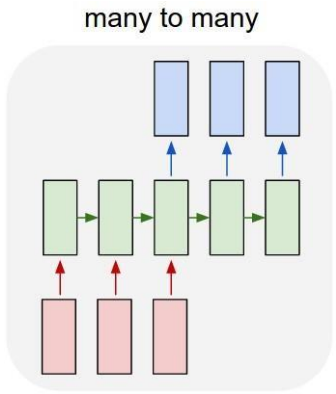
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Types of Task Dealing with Sequential Data

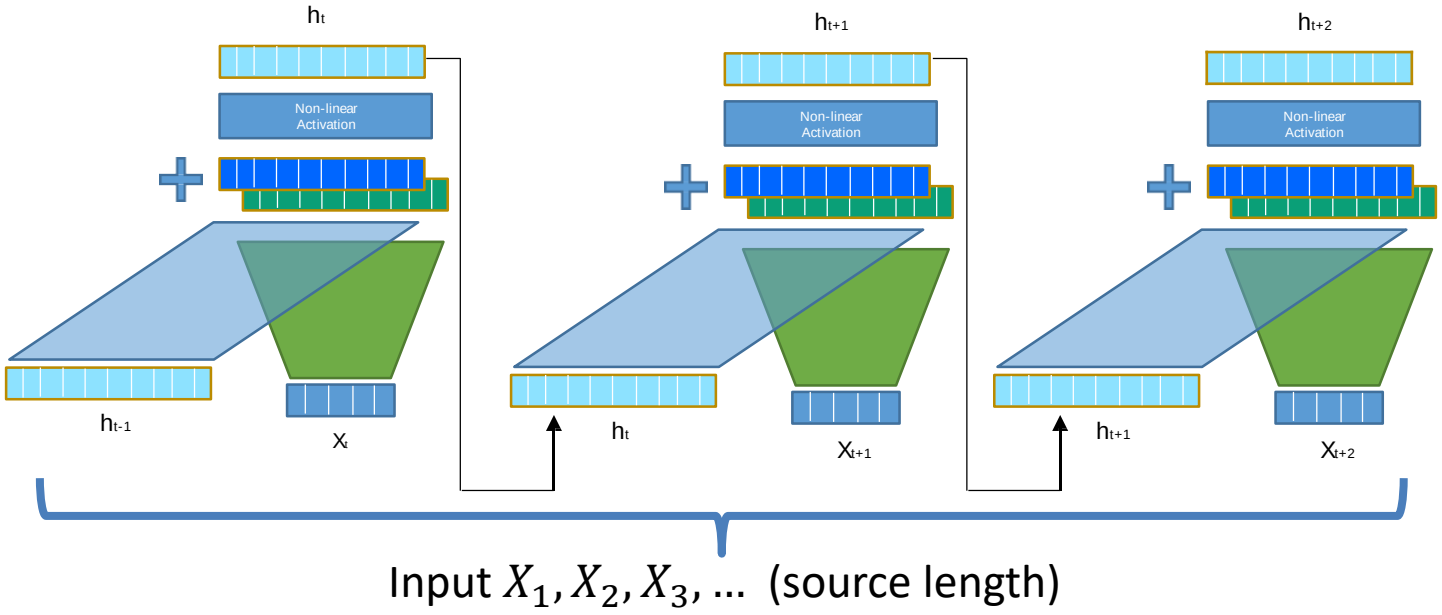


http://karpathy.github.io/2015/05/21/rnn-effectiveness/

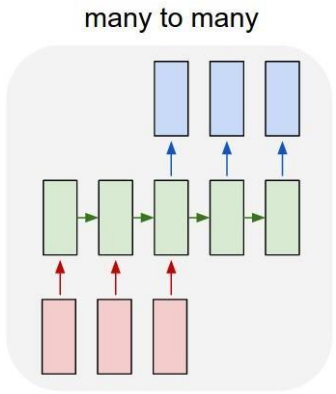
Recurrent Neural Network



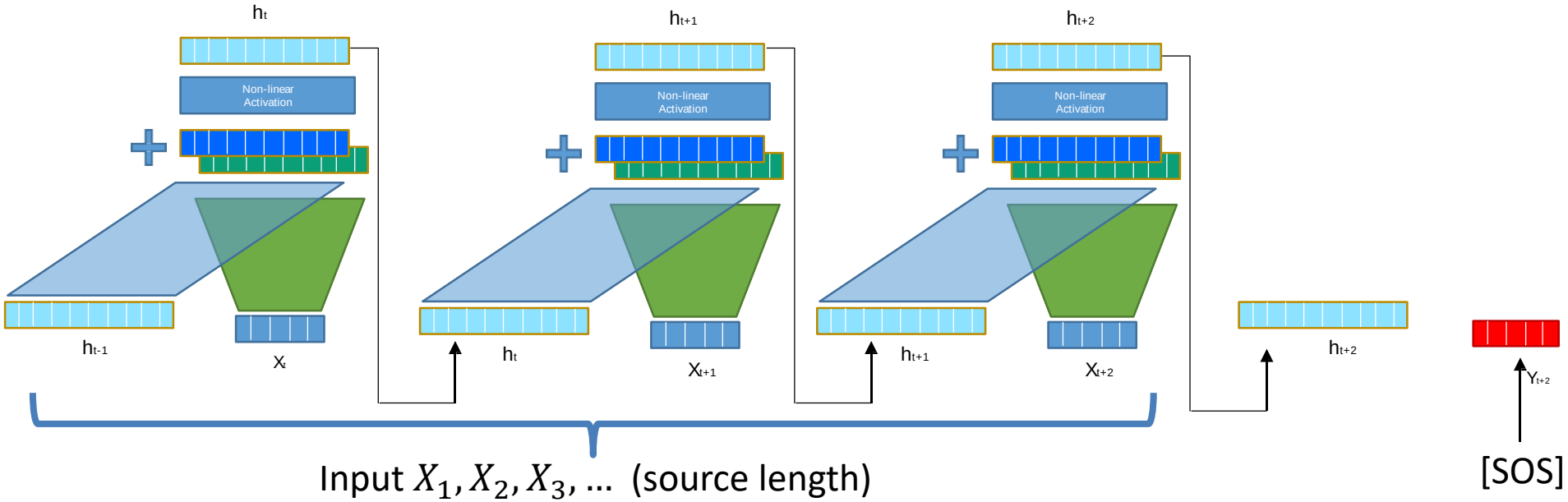
1. Run through **source inputs** to the encoder model.



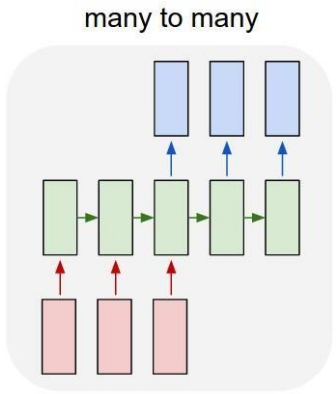
Recurrent Neural Network



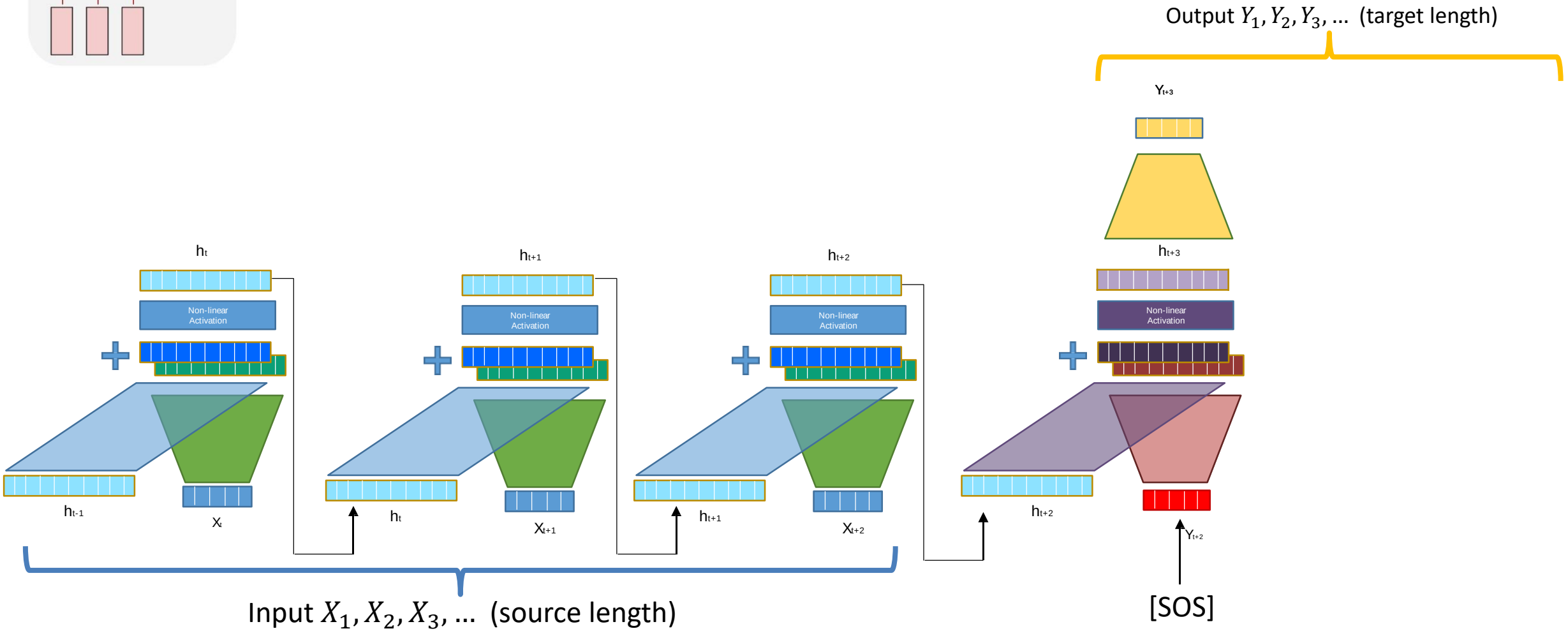
1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.



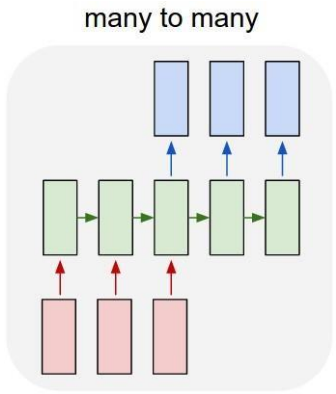
Recurrent Neural Network



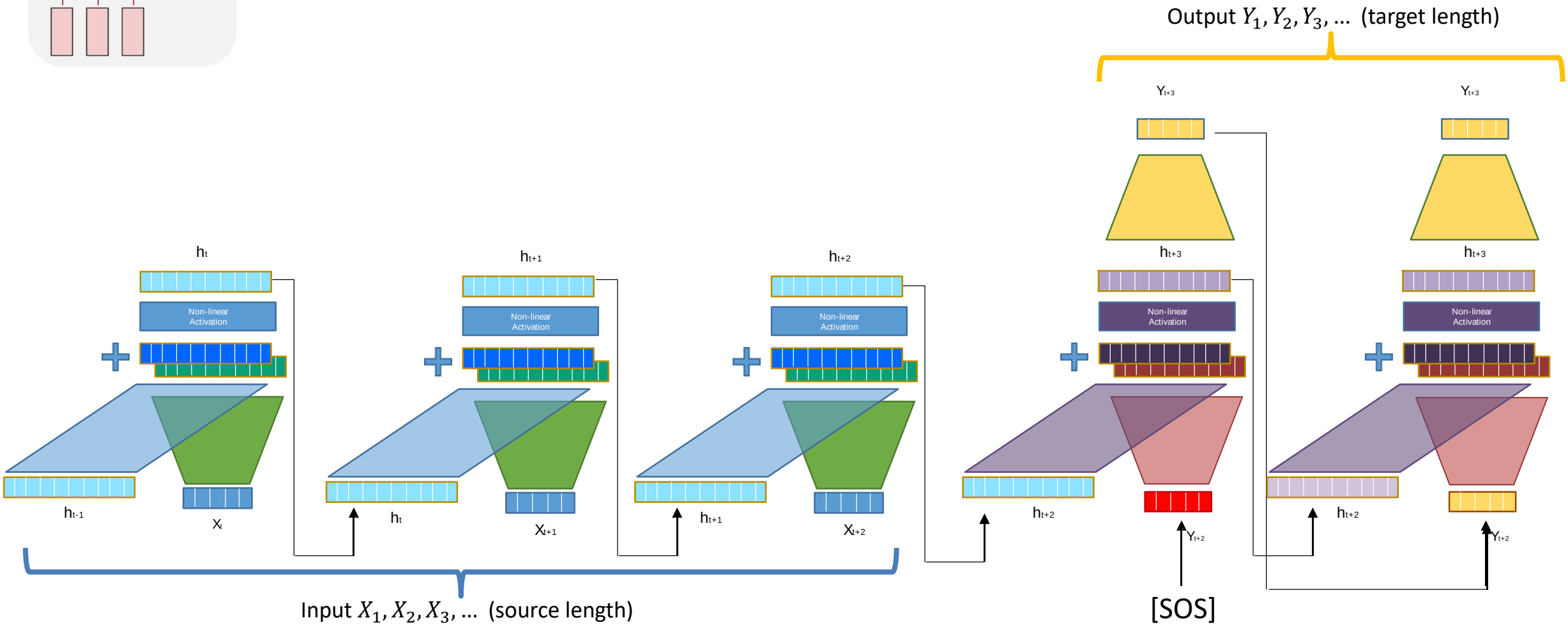
1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.



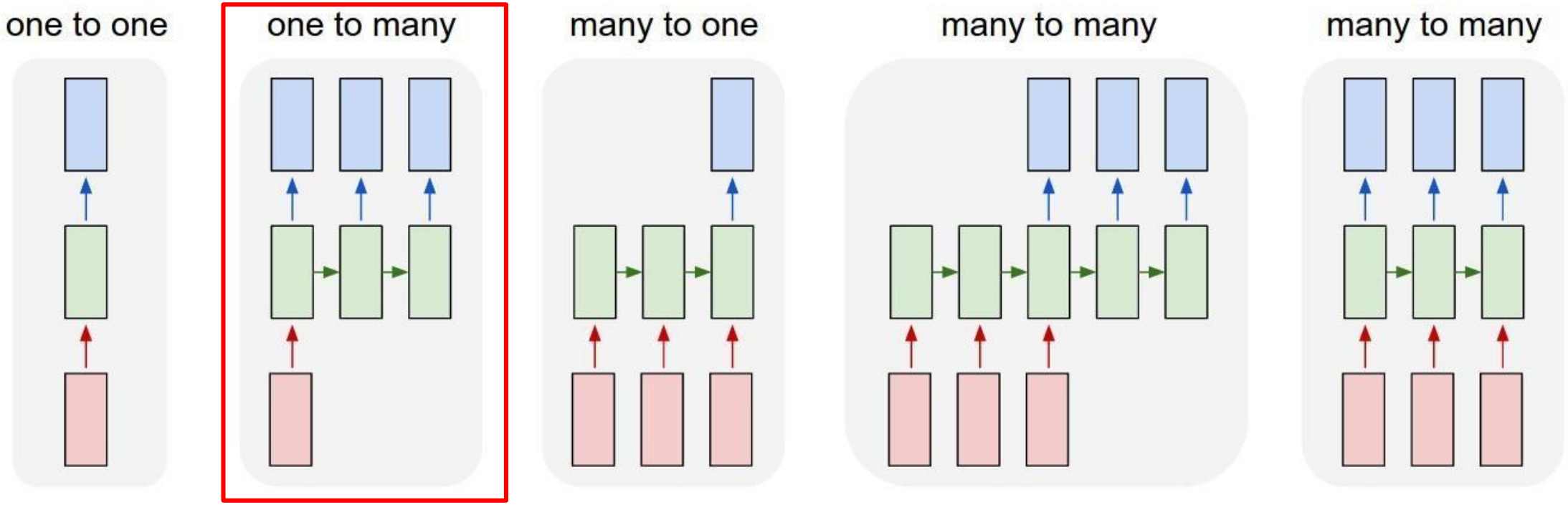
Recurrent Neural Network



1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.
4. By using for iteration, generate **next output token** until terminate condition.

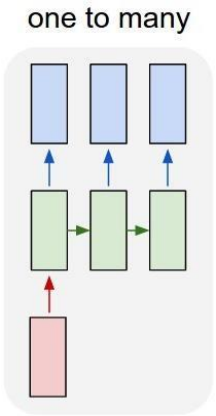


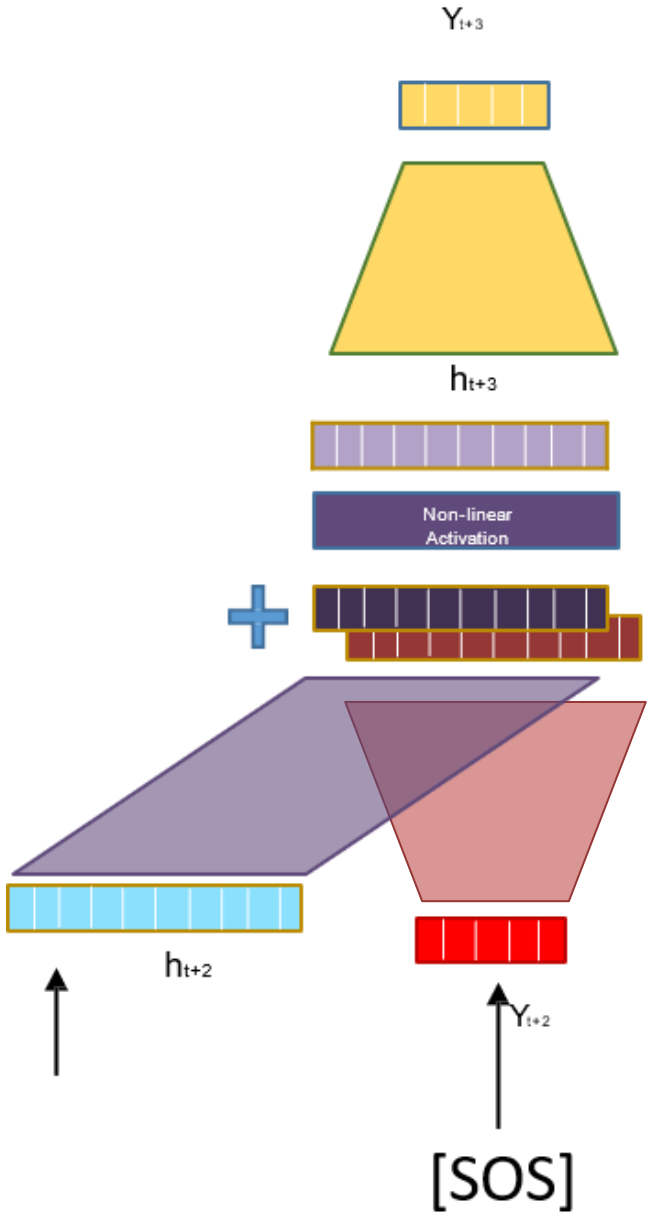
Types of Task Dealing with Sequential Data



http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Recurrent Neural Network

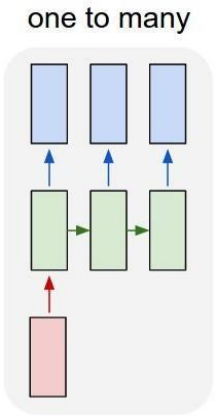


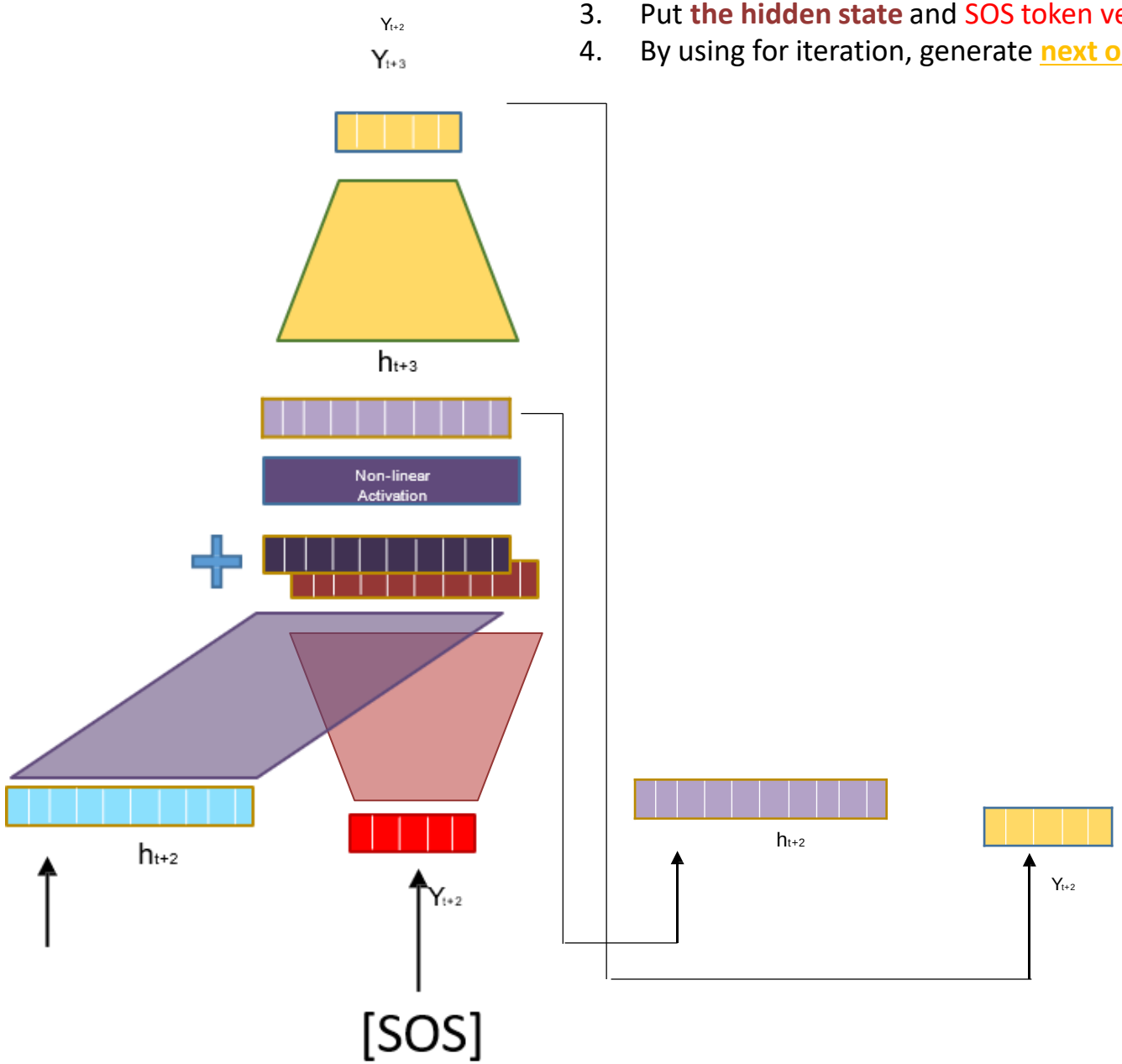


1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.
4. By using for iteration, generate **next output token** until terminate condition.

Recurrent Neural Network

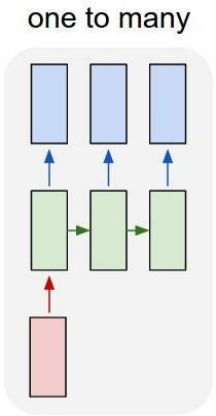
1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.
4. By using for iteration, generate **next output token** until terminate condition.

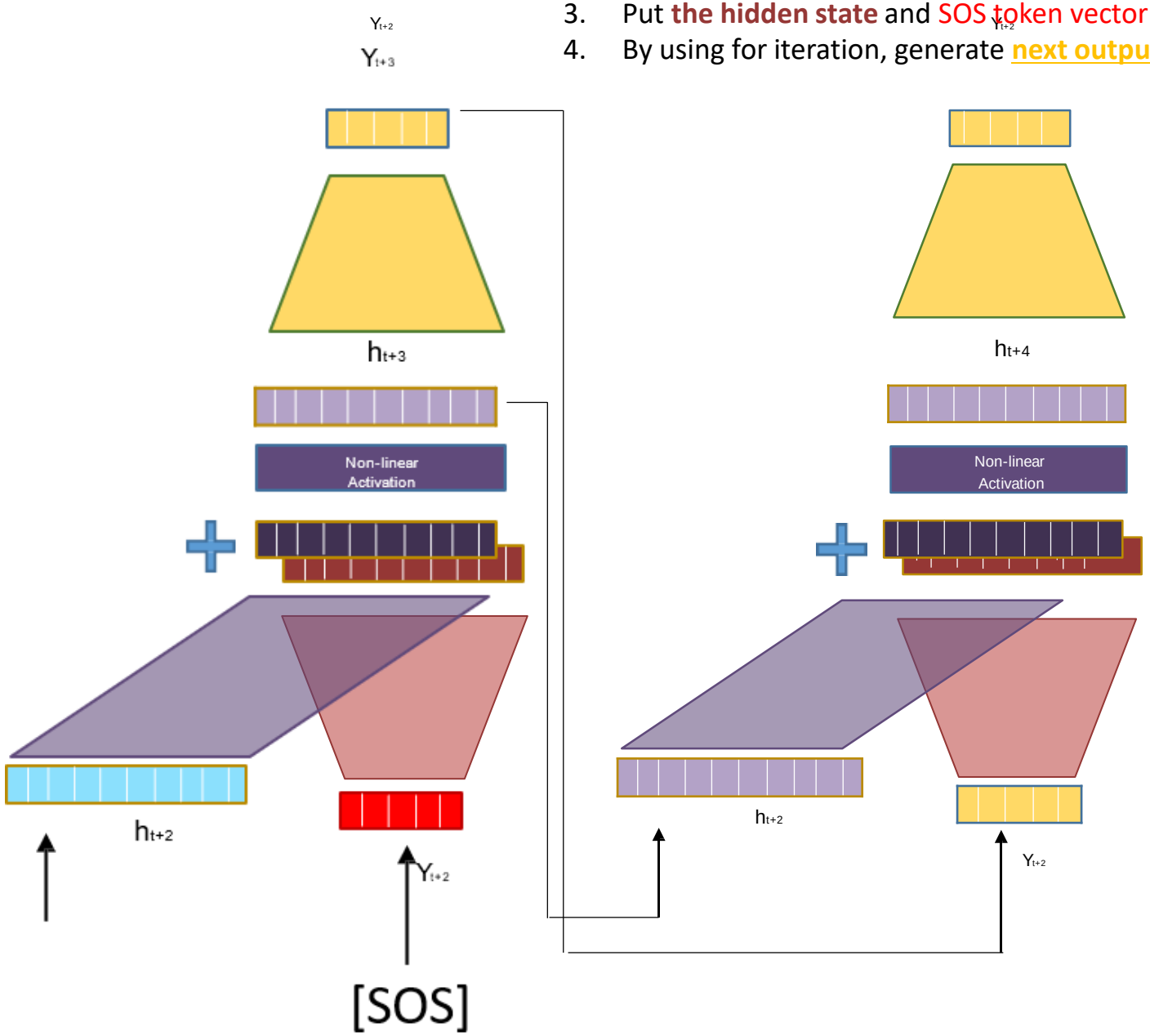




Recurrent Neural Network

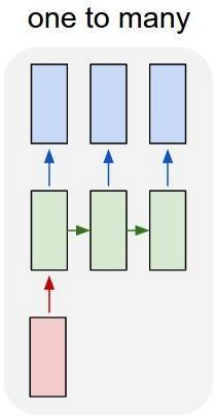
1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.
4. By using for iteration, generate **next output token** until terminate condition.

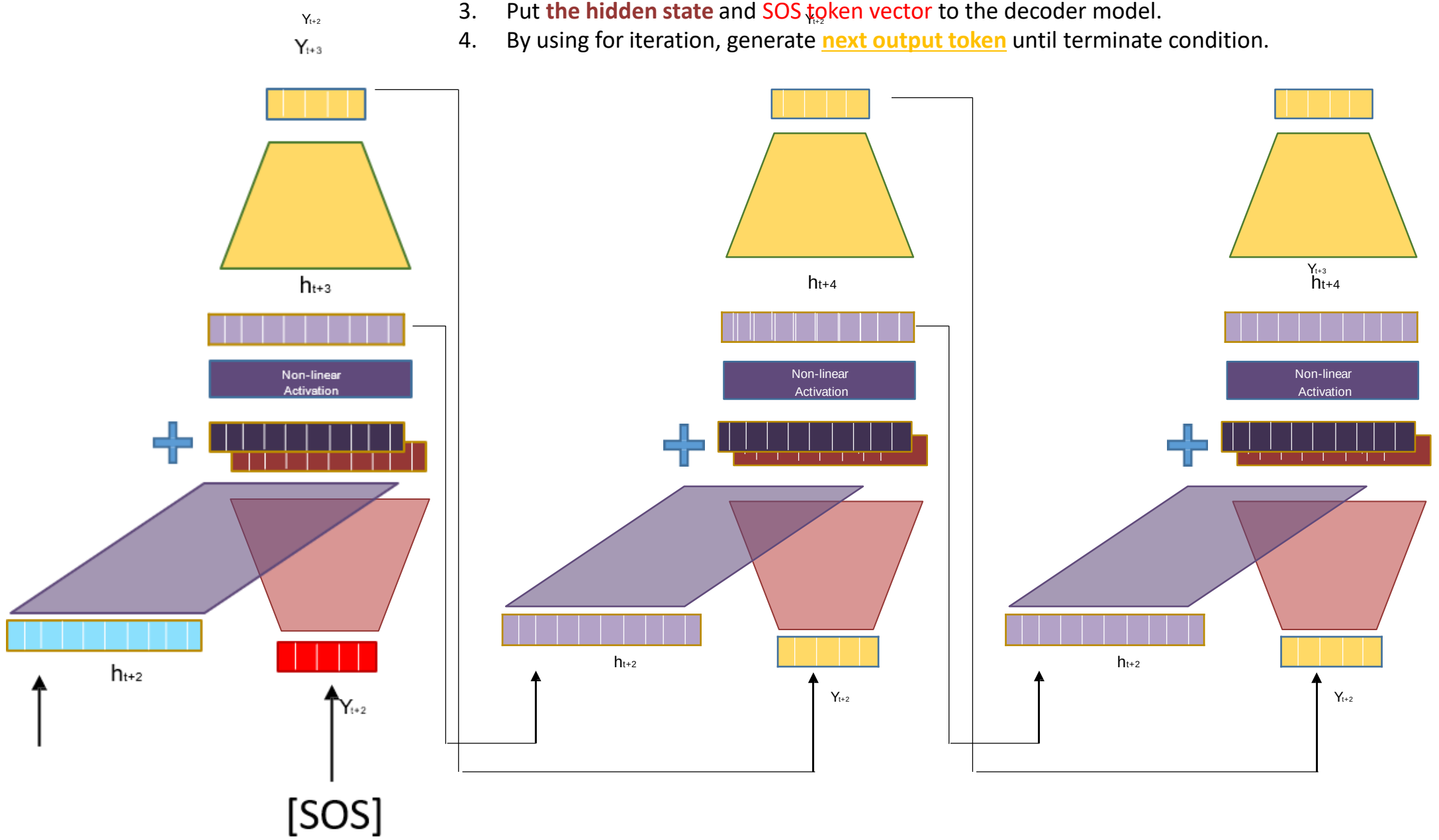




Recurrent Neural Network

1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.
4. By using for iteration, generate **next output token** until terminate condition.





Recurrent Neural Network

1. Run through **source inputs** to the encoder model.
2. Set **the last encoder hidden state** to **the first decoder hidden state**, create a **SOS token vector**.
3. Put **the hidden state** and SOS token vector to the decoder model.
4. By using for iteration, generate **next output token** until terminate condition.