Types of RNN

Types of RNN

1. One-to-Many

2. Many-to-One

3. Many-to-Many

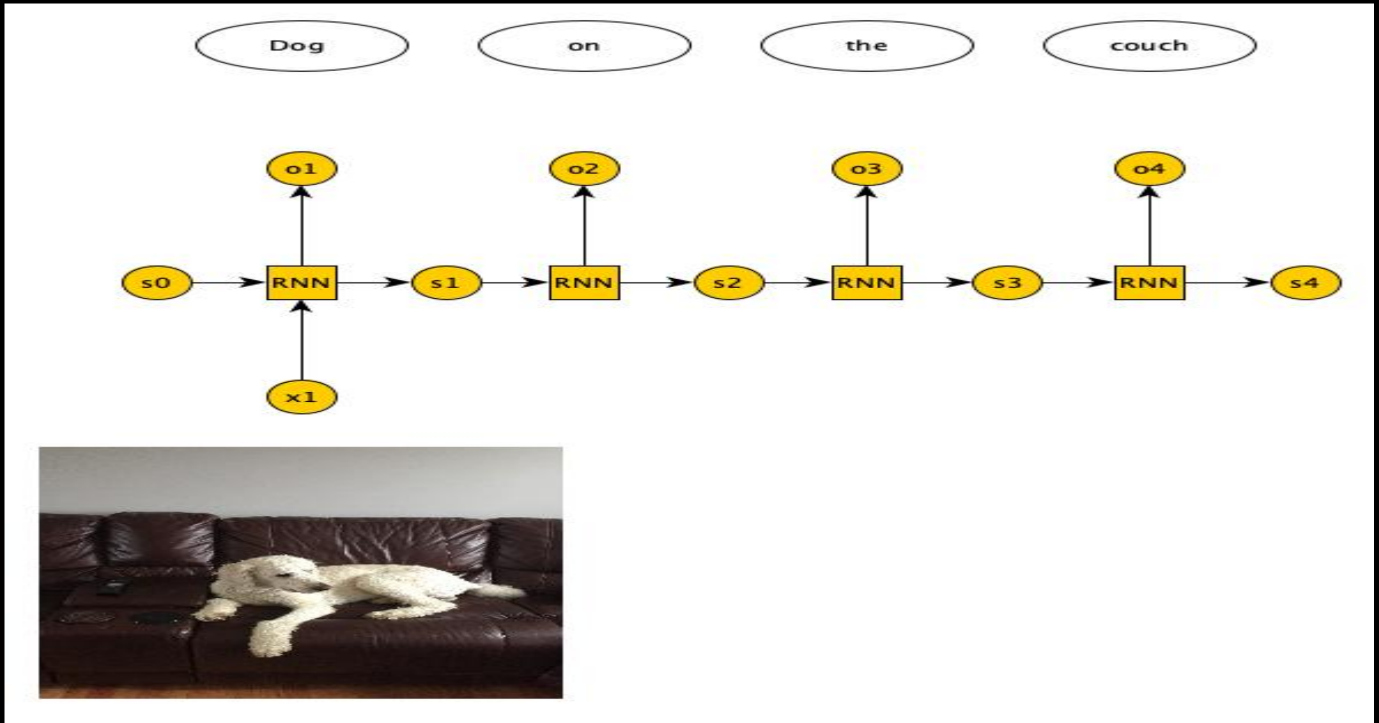
One-to-Many

- A single input (or fixed-length vector) generates a sequence of outputs.

Image Captioning

1. CNN encodes an image into a feature vector.
2. An RNN decoder takes that vector as its initial hidden state and emits a word at each time step: “A” → “dog” → “running” → … → <END>.

One-to-Many



https://training.galaxyproject.org/training-material/topics/statistics/tutorials/RNN/slides-plain.html

Many-to-One

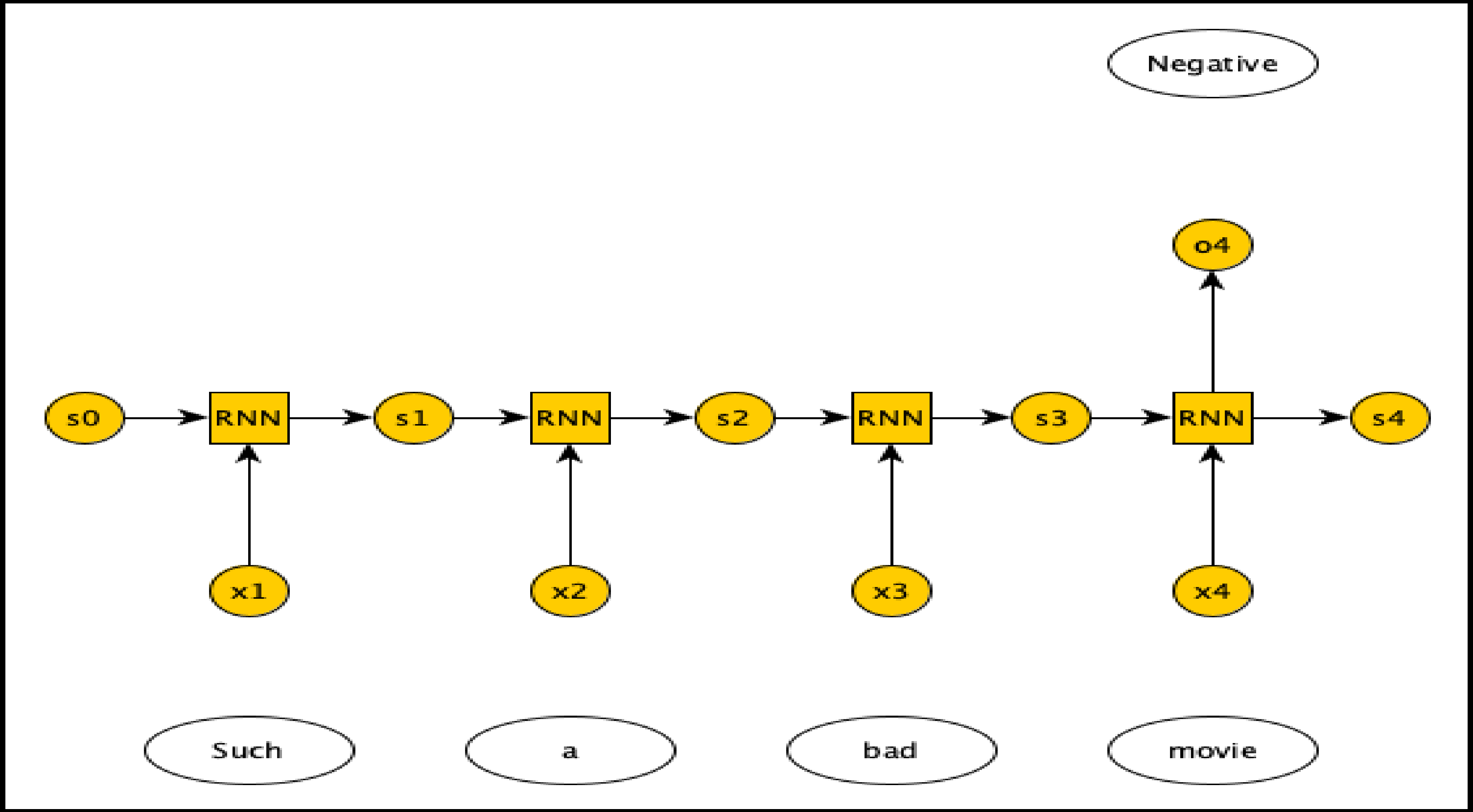
- A whole input sequence is read in; after the final time step, the model emits a single output.
- You unroll the RNN over all timesteps, then apply (e.g.) a dense layer to the last hidden state.

Many-to-One

Example:

- **Sentiment Classification**
  - Input: the word sequence of a movie review.
  - Output: one scalar (e.g. positive vs. negative) based on the final hidden state.
- **Sequence Regression**
  - Predicting tomorrow’s stock price from the past week’s closing prices.

Many-to-One



https://training.galaxyproject.org/training-material/topics/statistics/tutorials/RNN/slides-plain.html

Many-to-Many

- **Synchronized (Output length = Input length)**

Definition: One output for each input time step.

Example:

- **Part-of-Speech Tagging**
  - At each word in the sentence, you tag it as noun/verb/etc.
- **Named-Entity Recognition**
  - Emit a label for every token (“O”, “B-PERSON”, “I-ORG”, …).

Many-to-Many

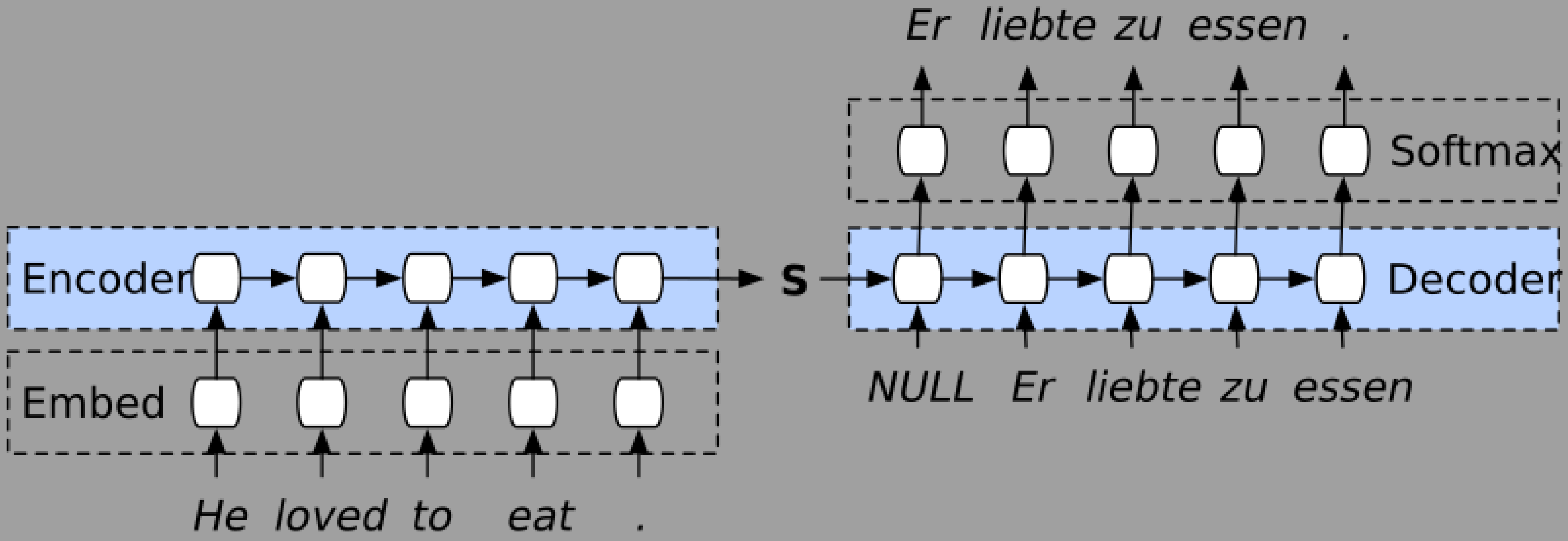
- **Unsynchronized (Output length ≠ Input length)**

Definition: You read the entire input sequence, then generate an output sequence (of different length).

Example:

- **Machine Translation**
  - **Encoder** RNN reads all words of the source sentence.
  - **Decoder** RNN then produces the target sentence one word at a time.

Many-to-Many


Er liebte zu essen .
Softmax
Encoder
S
Decoder
Embed
NULL Er liebte zu essen
He loved to eat .