

Recurrent Neural Networks



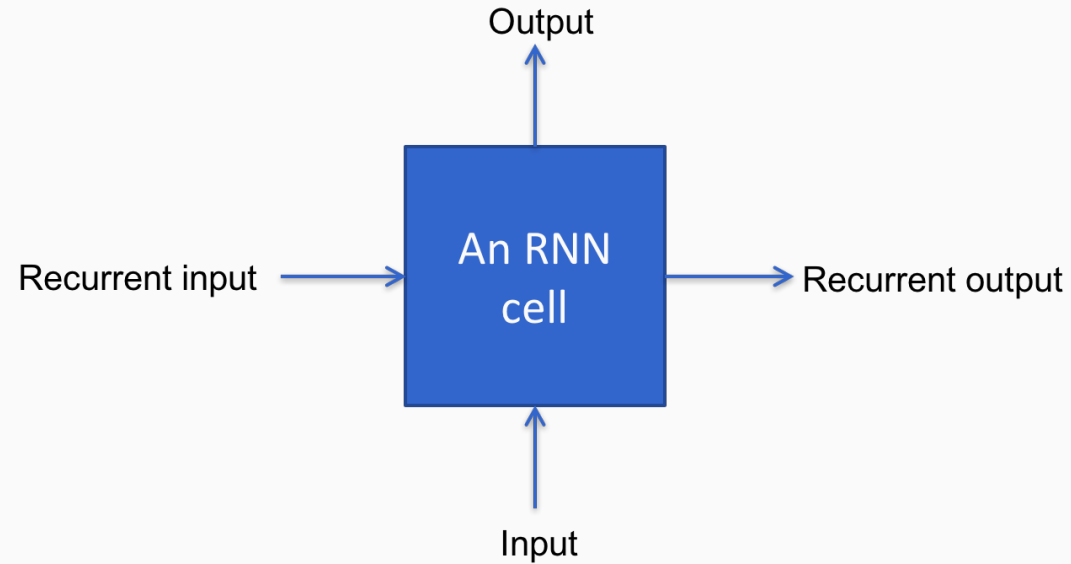
Overview

1. Modeling sequences
2. Recurrent neural networks: An abstraction
3. Usage patterns for RNNs
4. BiDirectional RNNs
5. A concrete example: The Elman RNN
6. The vanishing gradient problem
7. Long short-term memory units

Overview

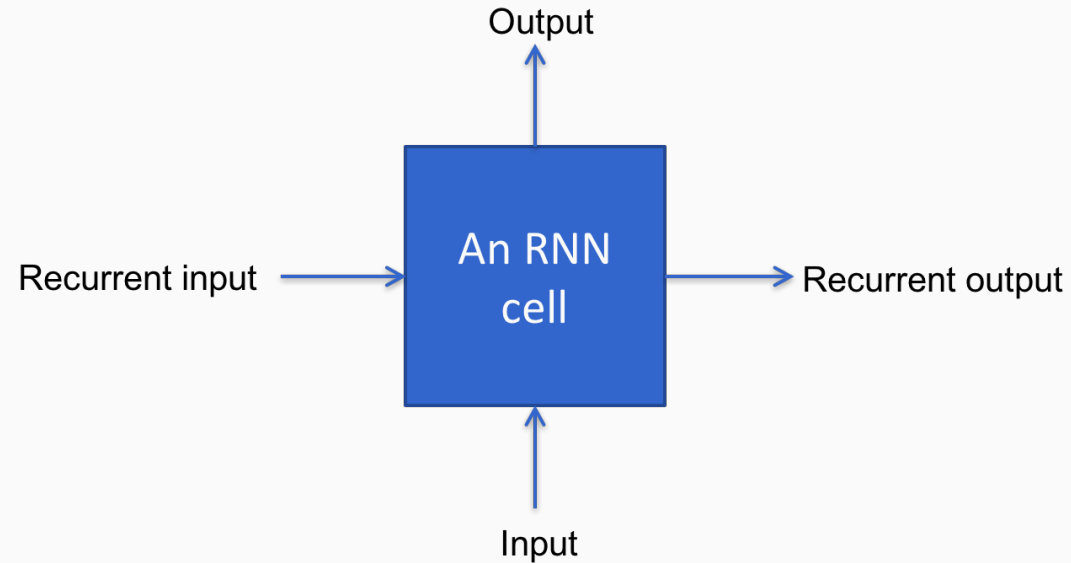
1. Modeling sequences
2. Recurrent neural networks: An abstraction
3. [Usage patterns for RNNs](#)
4. BiDirectional RNNs
5. A concrete example: The Elman RNN
6. The vanishing gradient problem
7. Long short-term memory units

What can we do with such an abstraction?



1. The **encoder**: Convert a sequence into a feature vector for subsequent classification
2. A **generator**: Produce a sequence using an initial state
3. A **transducer**: Convert a sequence into another sequence
4. A **conditional generator** (or an **encoder-decoder**): Combine 1 and 2

What can we do with such an abstraction?

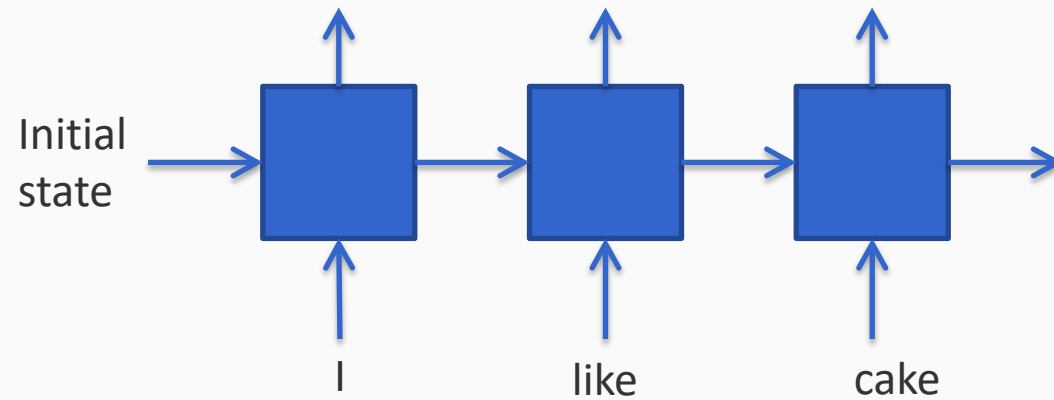


1. The **encoder**: Convert a sequence into a feature vector for subsequent classification
2. A **generator**: Produce a sequence using an initial state
3. A **transducer**: Convert a sequence into another sequence
4. A **conditional generator** (or an **encoder-decoder**): Combine 1 and 2

This set of operations also applies to other models for sequences. In particular, transformers.

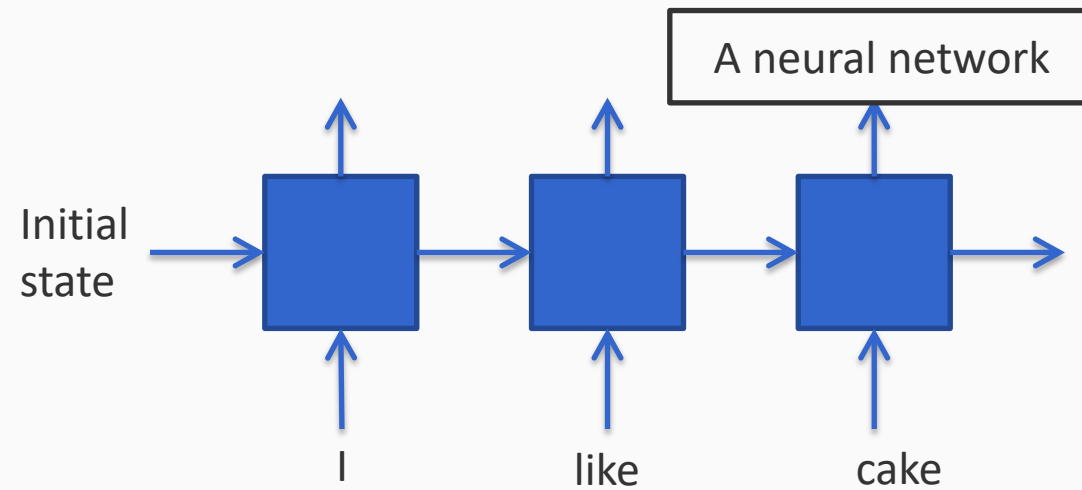
1. An Encoder

Convert a sequence into a feature vector for subsequent classification



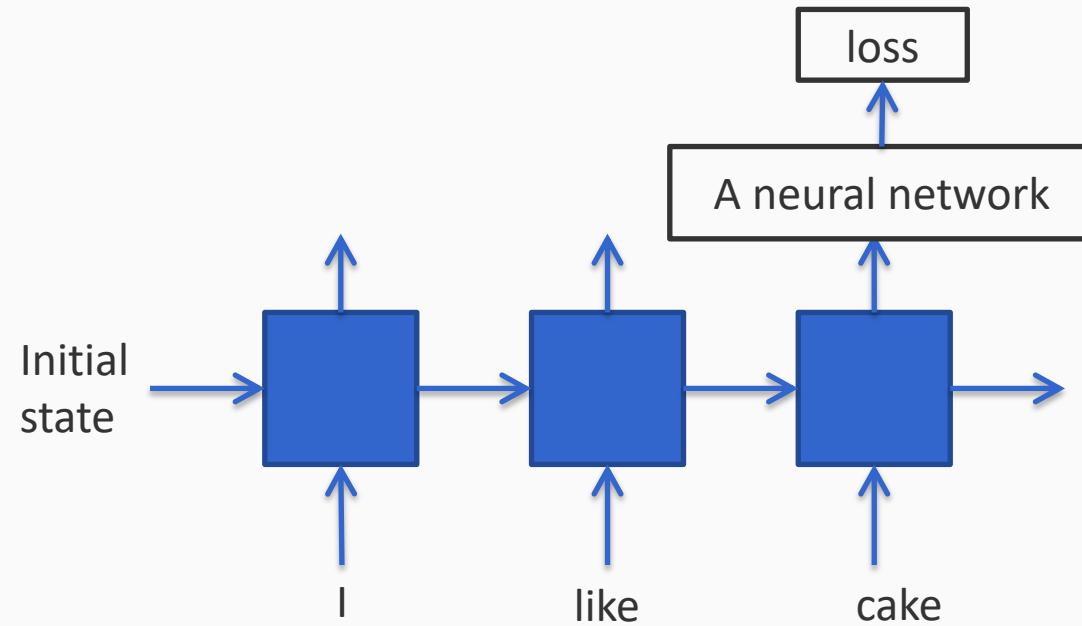
1. An Encoder

Convert a sequence into a feature vector for subsequent classification



1. An Encoder

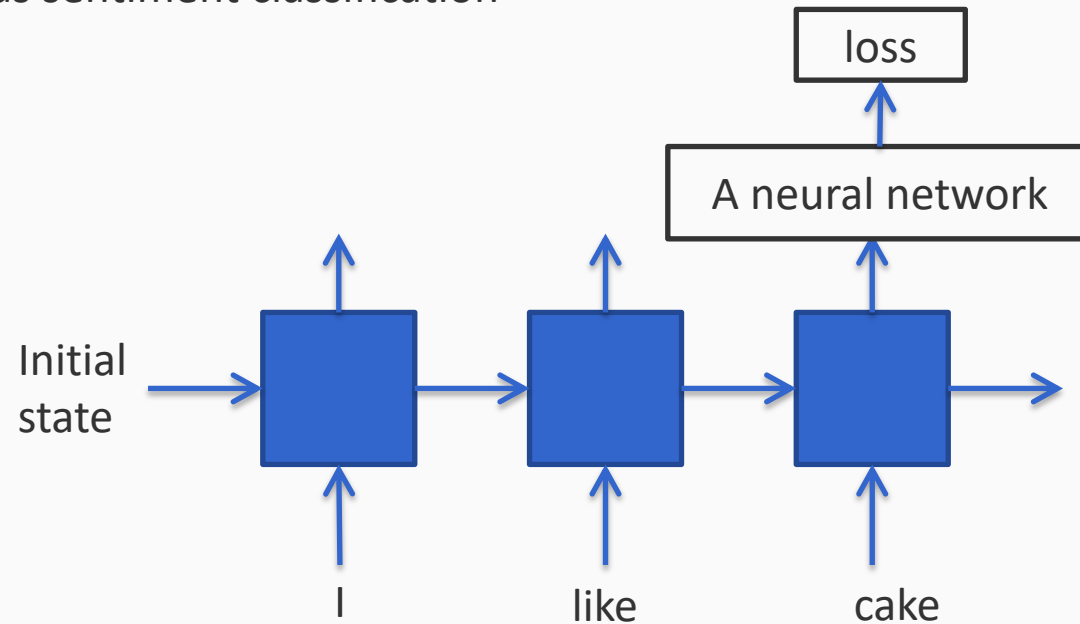
Convert a sequence into a feature vector for subsequent classification



1. An Encoder

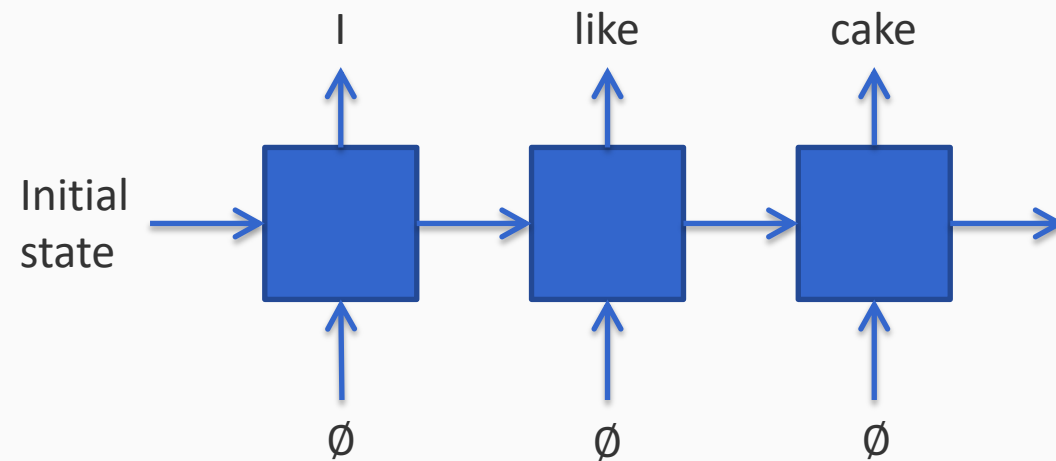
Convert a sequence into a feature vector for subsequent classification

Example: Encode a sentence or a phrase into a feature vector for a classification task such as sentiment classification



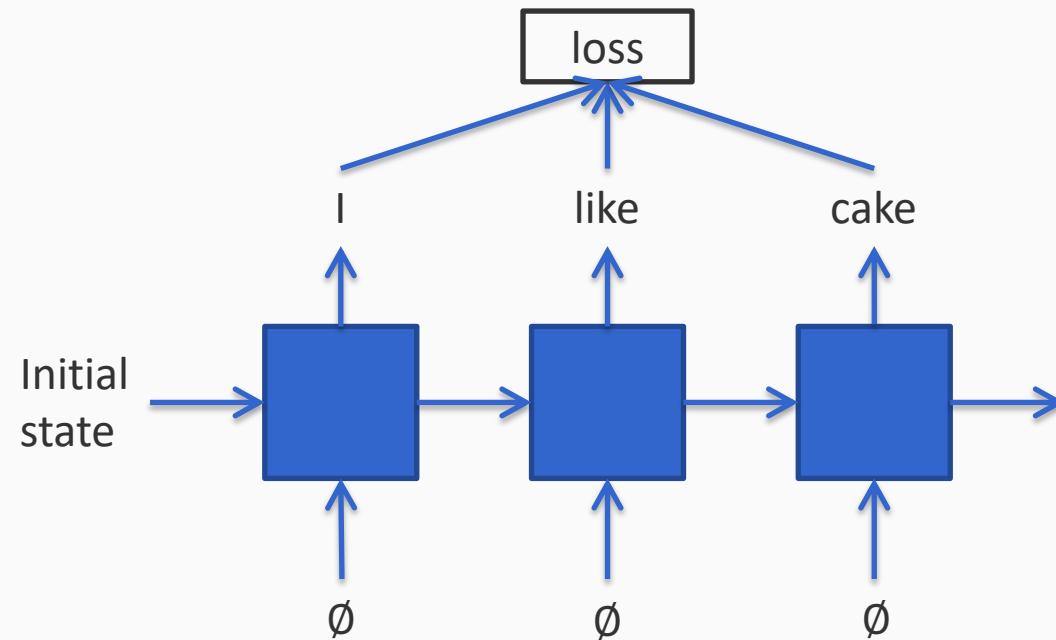
2. A Generator

Produce a sequence using an initial state



2. A Generator

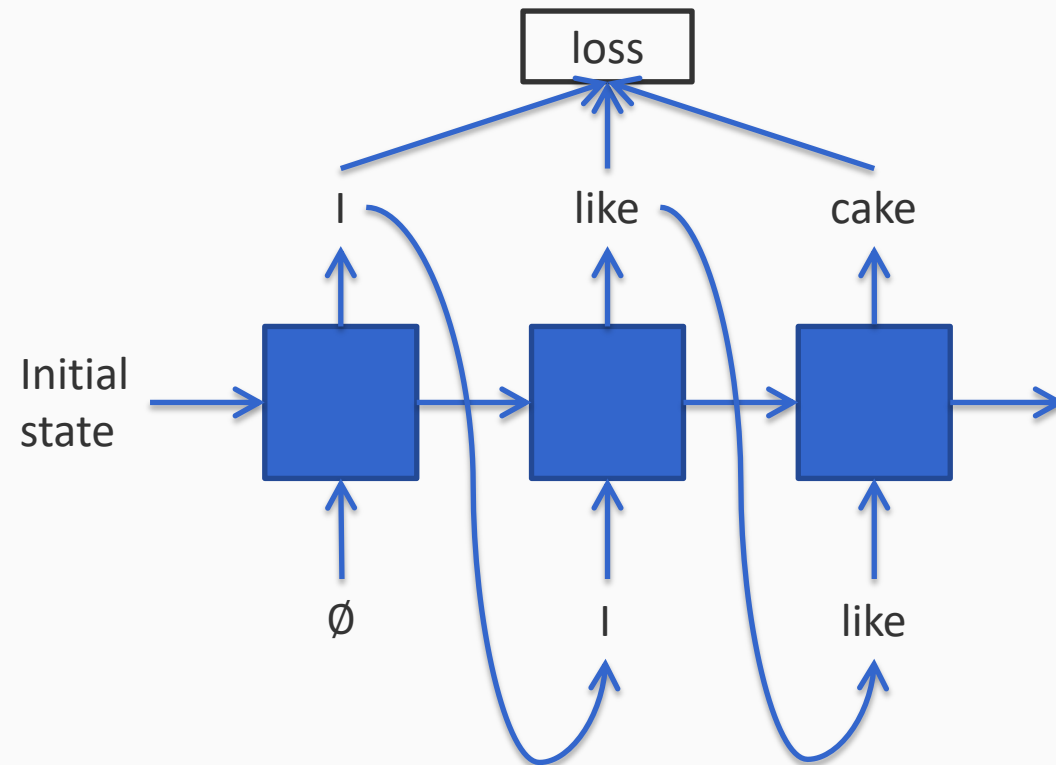
Produce a sequence using an initial state



2. A Generator

Produce a sequence using an initial state

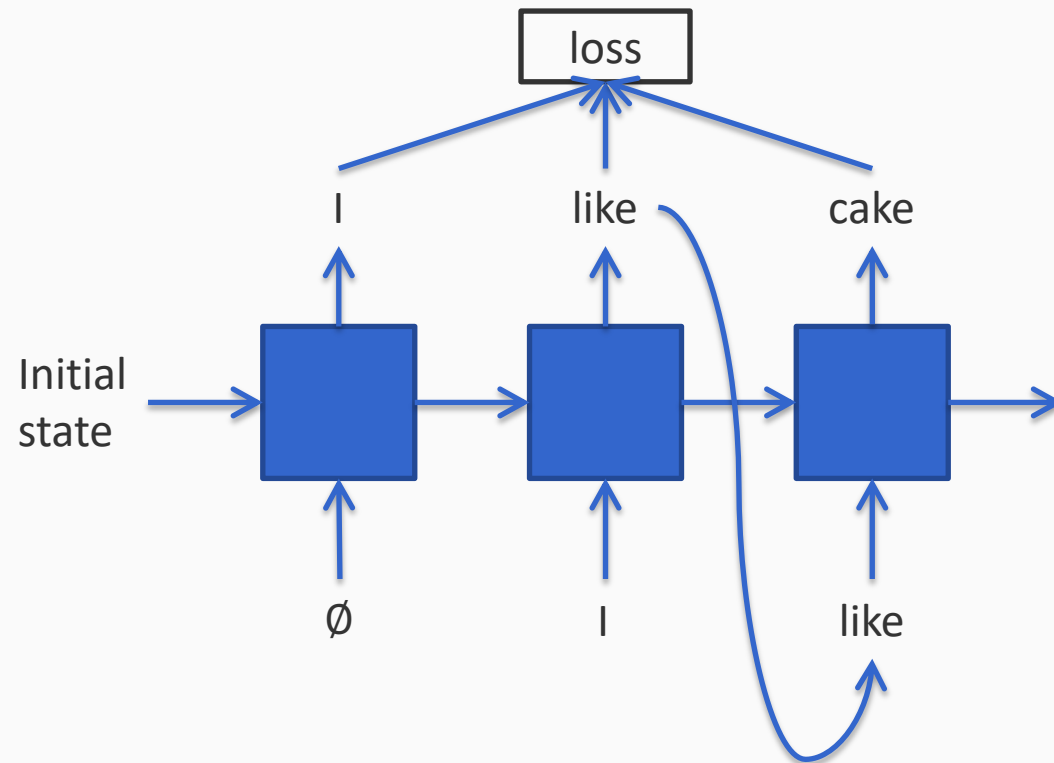
Maybe the previous output becomes the current input



2. A Generator

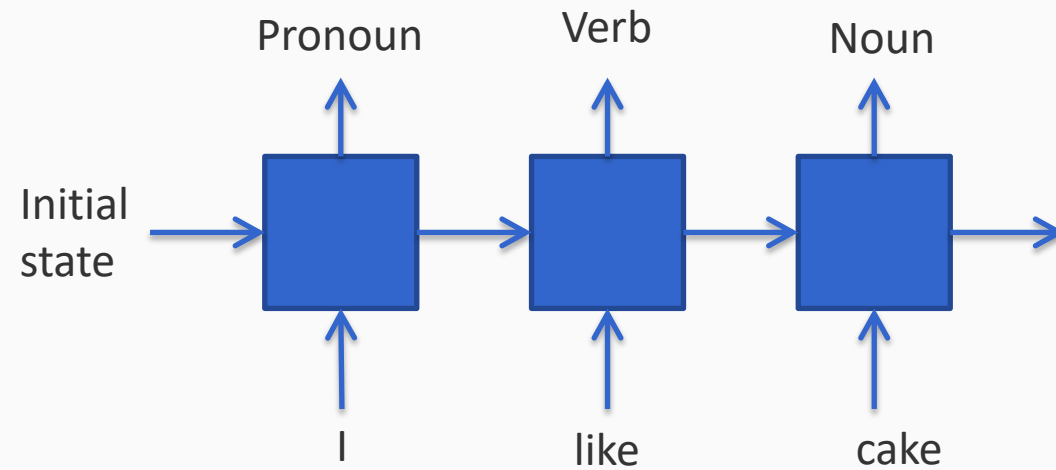
Produce a sequence using an initial state

Examples: Text generation tasks



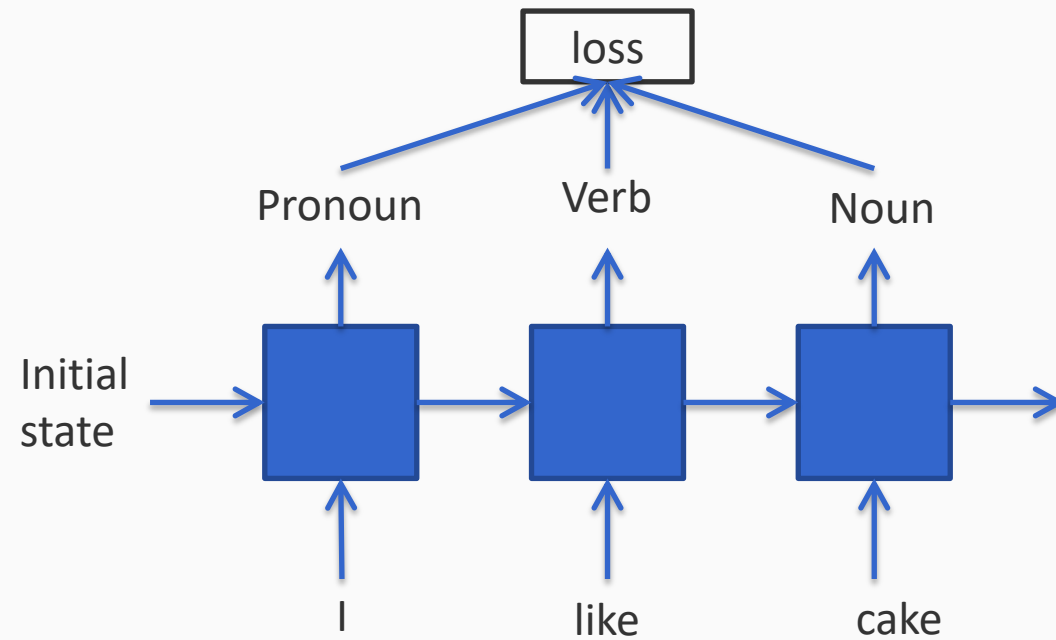
3. A Transducer

Convert a sequence into another sequence



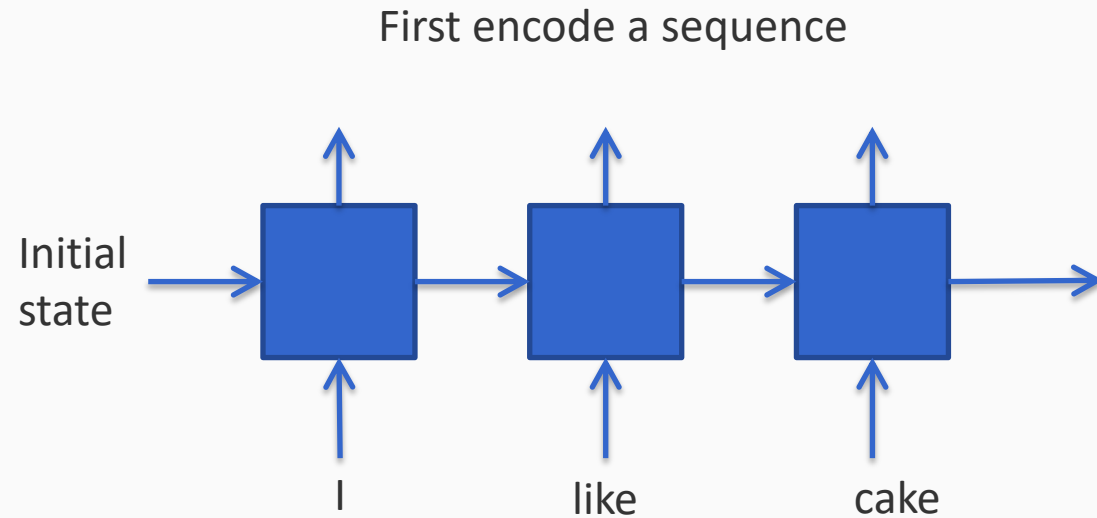
3. A Transducer

Convert a sequence into another sequence



4. Conditioned generator

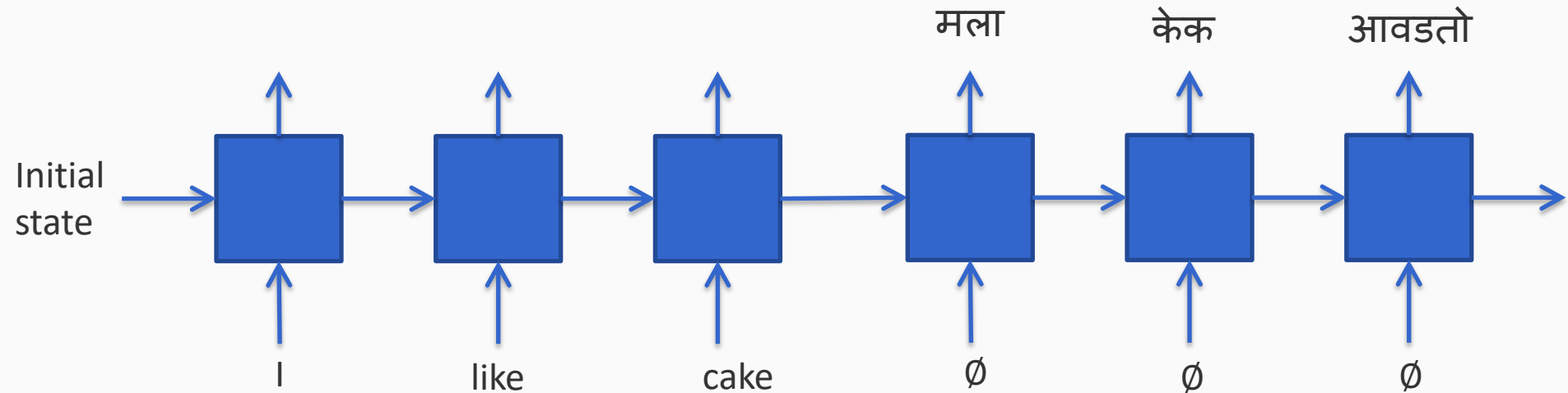
Or an encoder-decoder: First encode a sequence, then generate another one



4. Conditioned generator

Or an encoder-decoder: First encode a sequence, then generate another one

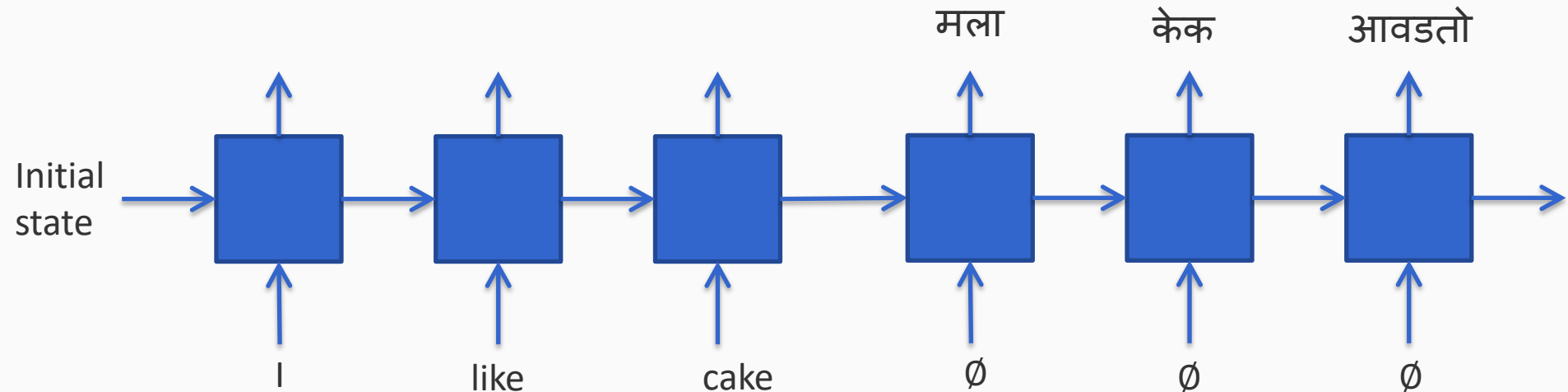
Then decode it to produce a different sequence



4. Conditioned generator

Or an encoder-decoder: First encode a sequence, then generate another one

Example: A building block for neural machine translation



Stacking RNNs

- A commonly seen usage pattern
- An RNN takes an input sequence and produces an output sequence
- The input to an RNN can itself be the output of an RNN – **stacked RNNs**, also called **deep RNNs**
- Two or more layers often seems to improve prediction performance