COMP34711

Deep Learning for NLP I

Tingting Mu
Department of Computer Science

Slides are based on CS224d: Deep Learning for Natural Language Processing

Deep Learning

- **Deep learning** (DL) and **neural network** (NN) are often used interchangeably. The subtle difference is that DL emphasises networks with higher number of layers.
- DL mostly refers to NN based techniques for building **end-to-end** systems, which take raw objects as the input.

input

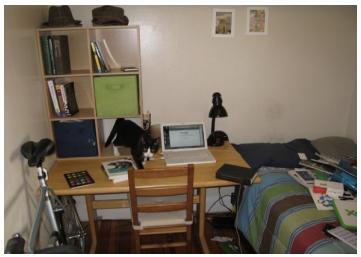
Question: how to push yourself to the limit during excercising?

Answer: try wearing a bandana and looking really cool and maybe you can ”push yourself to the limit” in a top gun kind of way. listen to some bon jovy music.

output

Decision: This is a correct answer to the question.

input



Question: what is on the bed?

output

Answer: books

- Deep learning is the state of the art technique for many NLP tasks.

Browse state-of-the-art in NLP: https://paperswithcode.com/area/natural-language-processing

- ***What can neural network do?***

Model Sequence Probability

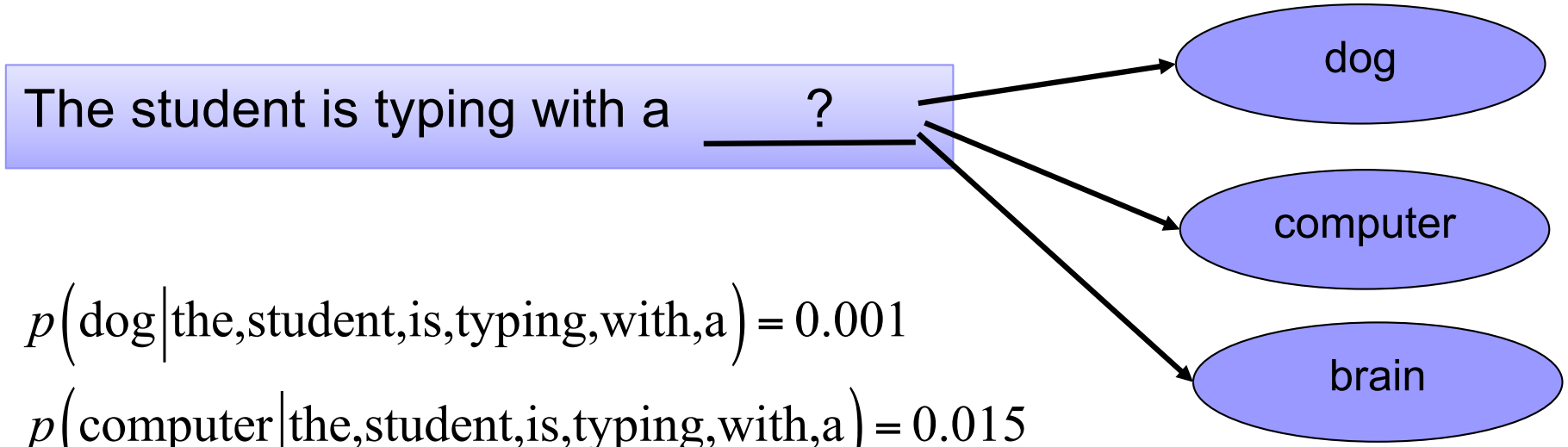
- Neural networks can be used to model probability distribution of a sequence.

$$p\left(x_k \middle| x_1, x_2, \ldots, x_{k-1}\right)$$

$$p\left(x_1, x_2, \ldots, x_{k-1}, x_k\right)$$

Example: Generate Text

- For example:



$$p\left(\text{dog}\middle|\text{the,student,is,typing,with,a}\right)=0.001$$

$$p\left(\text{computer}\middle|\text{the,student,is,typing,with,a}\right)=0.015$$

$$p\left(\text{brain}\middle|\text{the,student,is,typing,with,a}\right)=0.003$$

"Computer" is chosen with highest probability.

Sequence Representation and Classification

- Neural networks can be used to learn a representation vector of a sequence, and use it to classify the sequence.

$$f\left(x_1 \rightarrow x_2 \rightarrow \ldots, x_{k-1} \rightarrow x_k\right) = class$$

Example: Sentence Classification

- Sentiment analysis: Classify the sentiment of a sentence.

f(overall I enjoyed the movie a lot) = positive class

f(the quality of this knife is not great) = negative class

- Spam filtering.

f(from micro soft team… online security team)= spam class

From: "Microsoft Team" <inegon06@netscape.com>
Date: 7 October 2015 at 17:53:10 BST
To: <customerservice@outlook.com>
Subject: Avoid Suspension 2015!!!

Outlook

Dear Subscriber,

Your Microsoft account has been compromised.You must update it immediately or your account will be closed.

Click here to update

Sincerely,

Microsoft Online Security Team

Sequence Labelling

- Neural networks can be used to learn a representation vector for each state (element) in a sequence, and use it to predict the class label for each state.

$$f\left(x_1 \rightarrow x_2 \rightarrow \ldots, x_{k-1} \rightarrow x_k\right)$$
$$= class_1 \rightarrow class_2 \rightarrow \ldots, class_{k-1} \rightarrow class_k$$

Example: POS Tagging

- Part-of-speech tagging: Assign a word to a priorly defined lexical class.

f(the startled cat knocked over the vase)
= (DT JJ NN VBN IN DT NN)

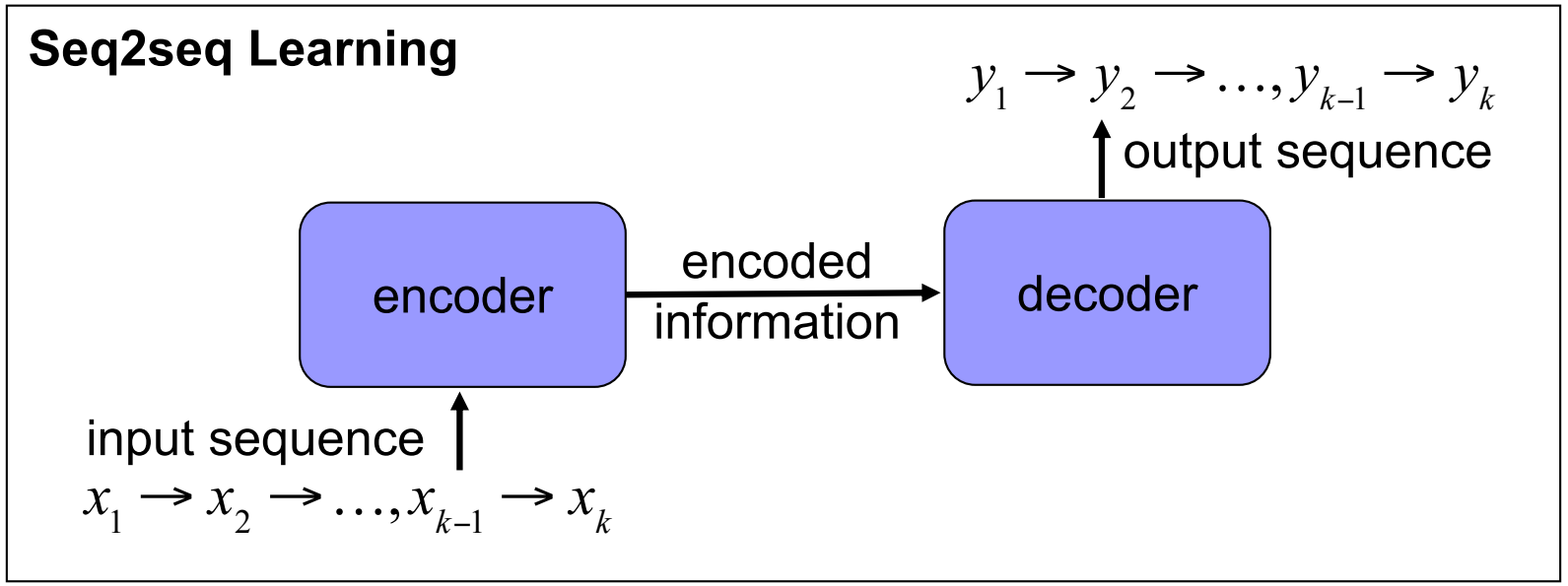
- Named entity recognition: Assign a word to a name class (e.g. in a name class or in no name class).

f(Fred showed Sue Mengqiu Huang's new painting)
= (PER O PER PER PER O O O)

Sequence to Sequence Learning

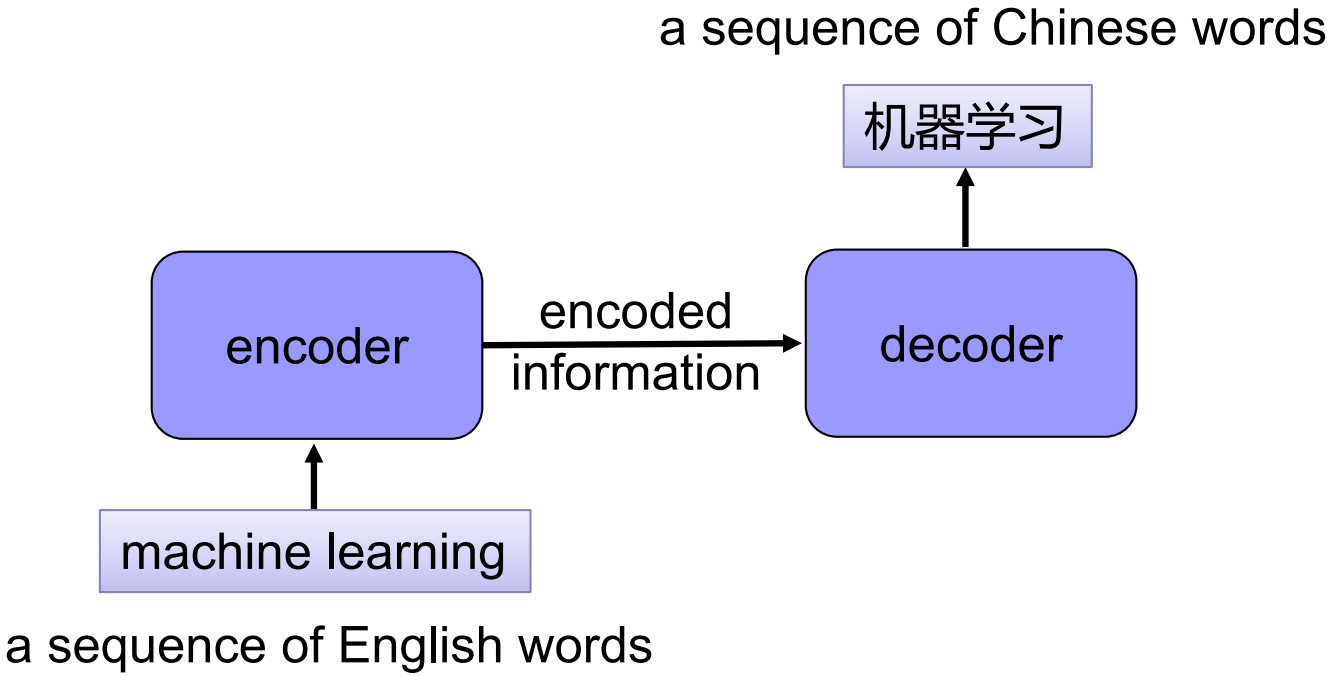
- Neural networks can be used to encode information in an input sequence (seq), and decode it to generate an output sequence (2seq).

$$f\left(x_1 \rightarrow x_2 \rightarrow \ldots, x_{k-1} \rightarrow x_k\right) = y_1 \rightarrow y_2 \rightarrow \ldots, y_{k-1} \rightarrow y_k$$



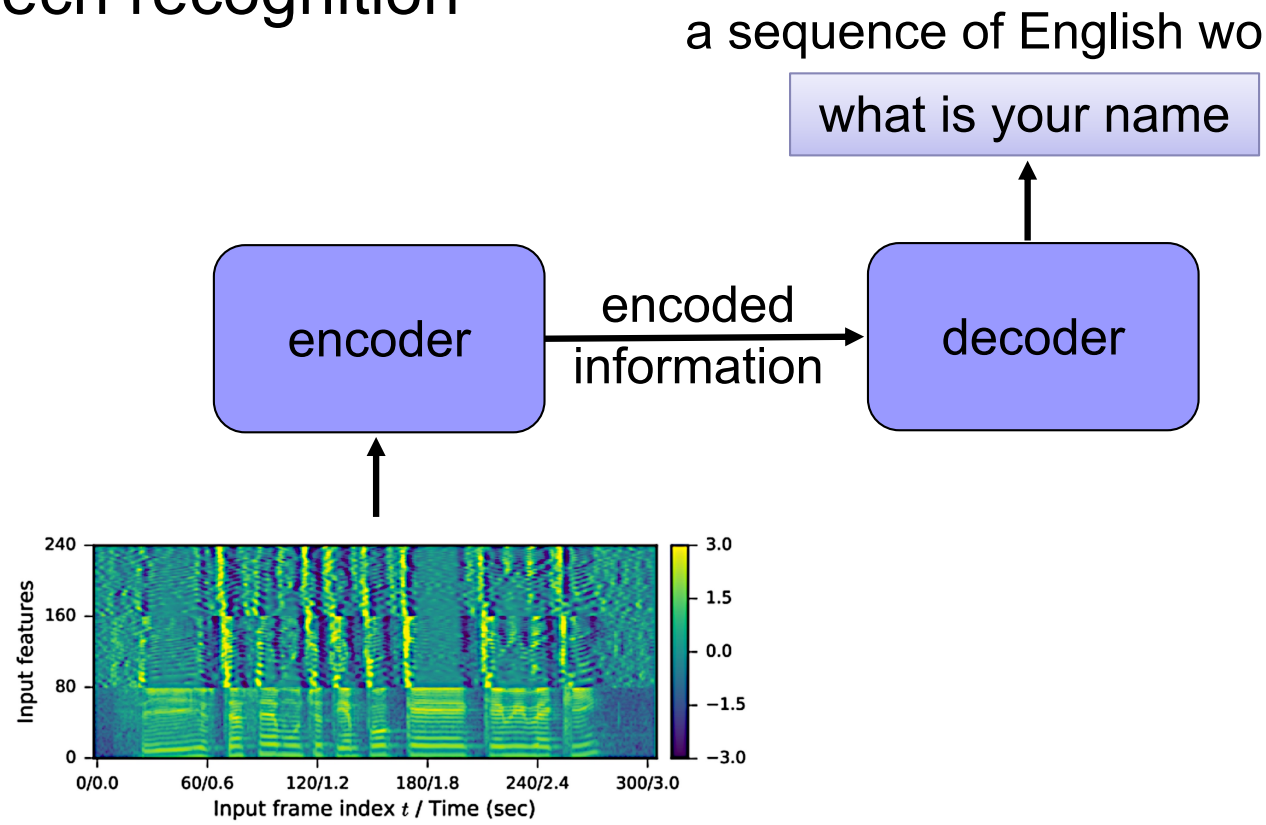
Examples

- Machine translation



Examples

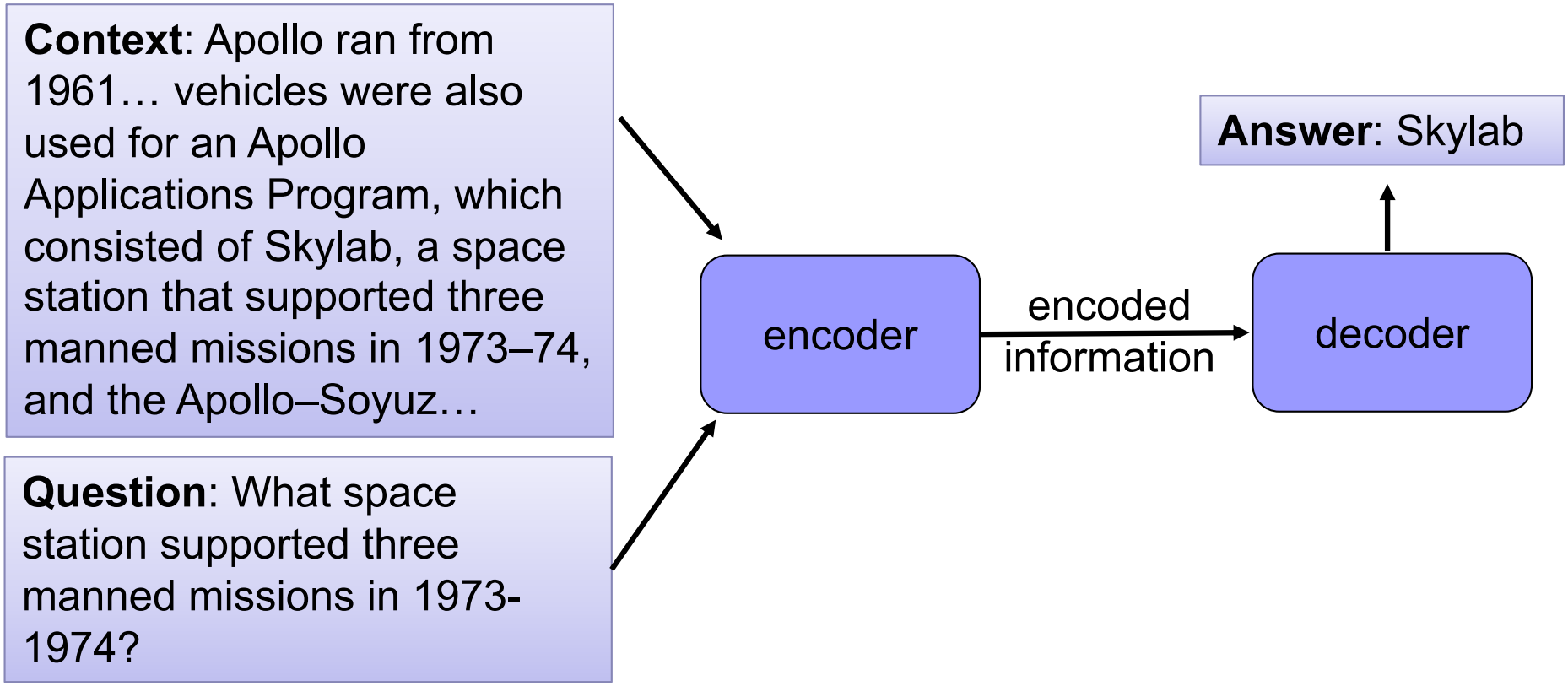
- Speech recognition



Feature figure is from https://www.arxiv-vanity.com/papers/1703.08581/

Examples

- Question answering



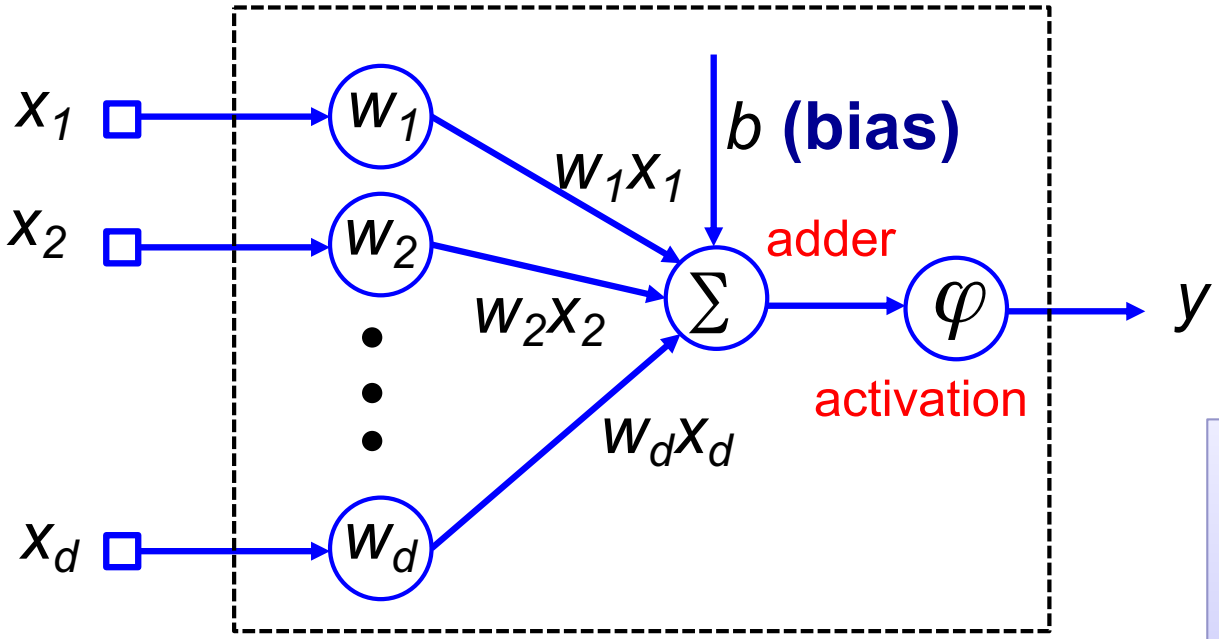
- ***Basic Neural Network Architectures***

  This part is taught in your year 2 Machine Learning course.

  Revisit the knowledge.

Single Neuron

- An artificial neuron: multiple inputs [x_1, x_2, …, x_d] and one output y.



neuron $y = \varphi\left(\sum_{i=1}^{d} w_i x_i + b\right)$

Given d input values, a single neuron has $d+1$ parameters. Training is the process of finding the best values of these $d+1$ parameters.

Typical Activation Functions

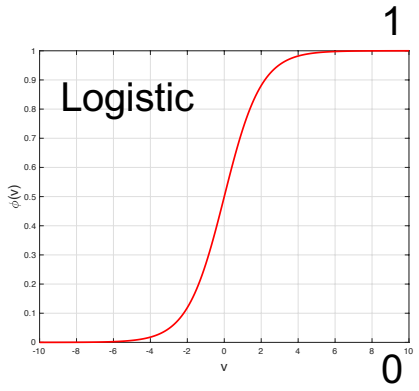
- Sigmoid function ("S"-shaped curve):

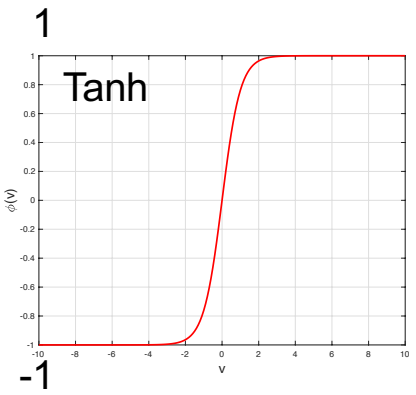
  Logistic function:

$$\varphi(v) = \frac{1}{1+\exp(-v)} \in (0,1)$$

  Hyperbolic tangent:

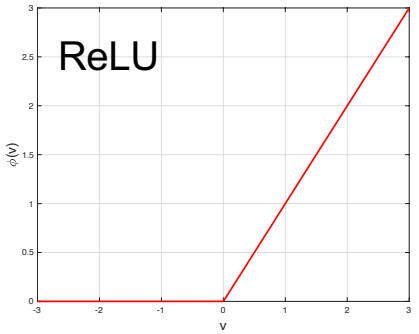
$$\varphi(v) = \tanh(v) = \frac{\exp(2v)-1}{\exp(2v)+1} \in (-1,+1)$$

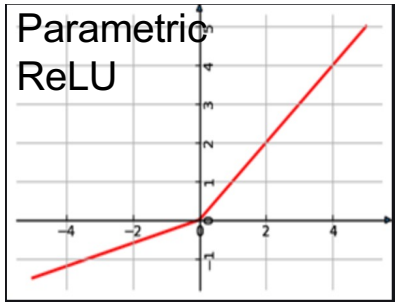




- Rectified linear unit (ReLU):

$$\varphi(v) = \begin{cases} v & if \quad v \geq 0 \\ 0 & if \quad v < 0 \end{cases}$$



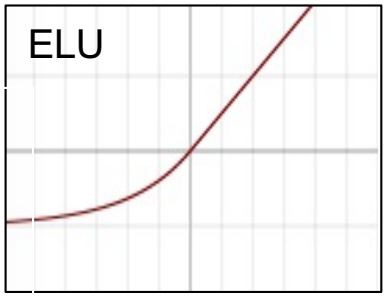


- Parametric ReLU (a=0.01 Leaky Relu)

$$\varphi(v) = \begin{cases} v & if \quad v \geq 0 \\ av & if \quad v < 0 \end{cases}$$

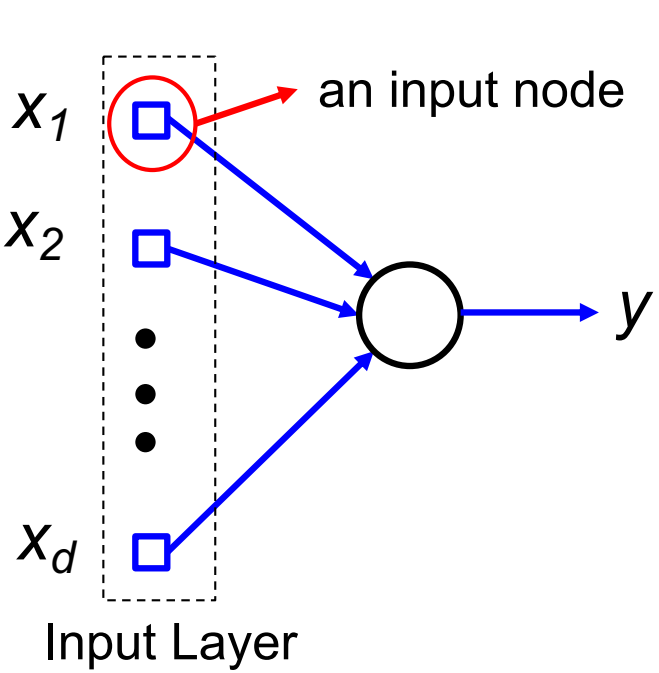
- ELU

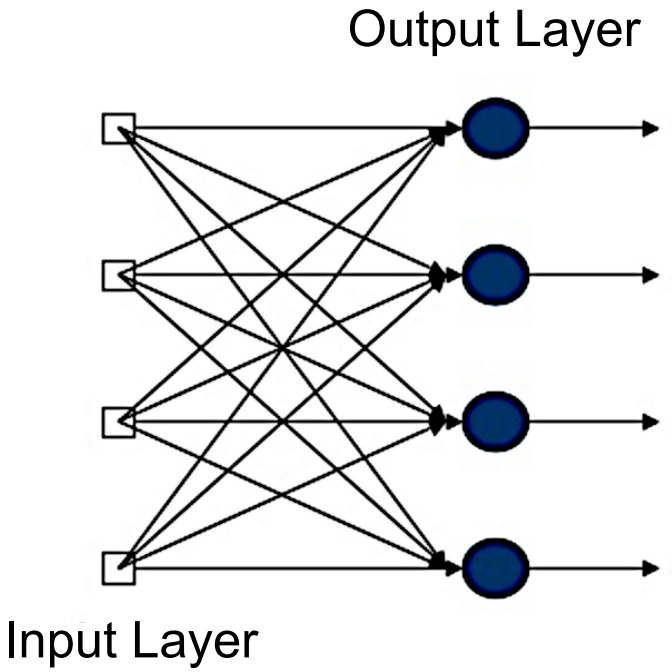
$$\varphi(v) = \begin{cases} v & if \quad v \geq 0 \\ a(e^{v}-1) & if \quad v < 0 \end{cases}$$



Single Layer Perceptron

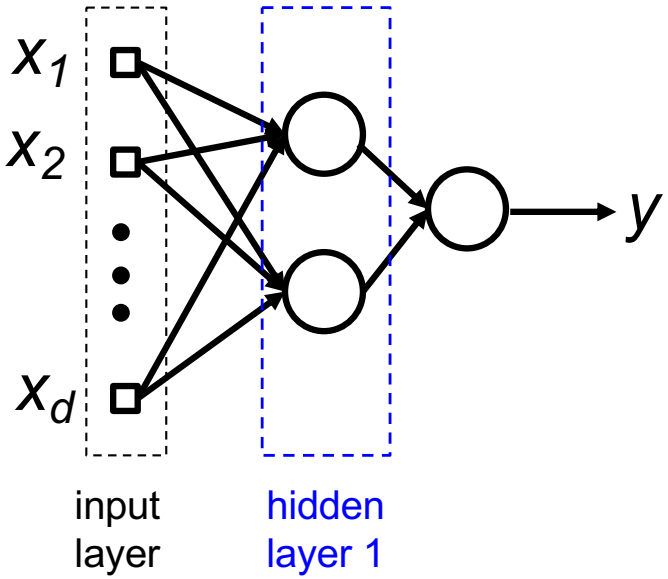
- A single layer perceptron has one input layer and one output layer.

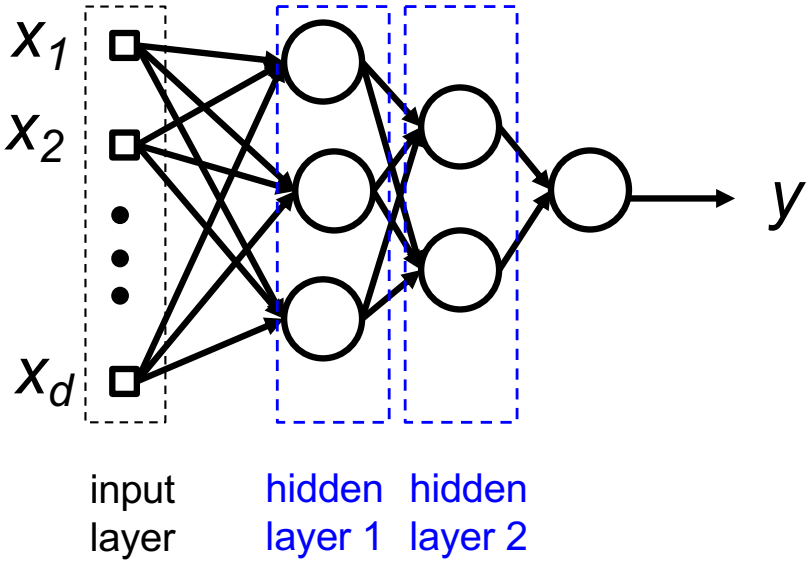




Adding Hidden Layers!

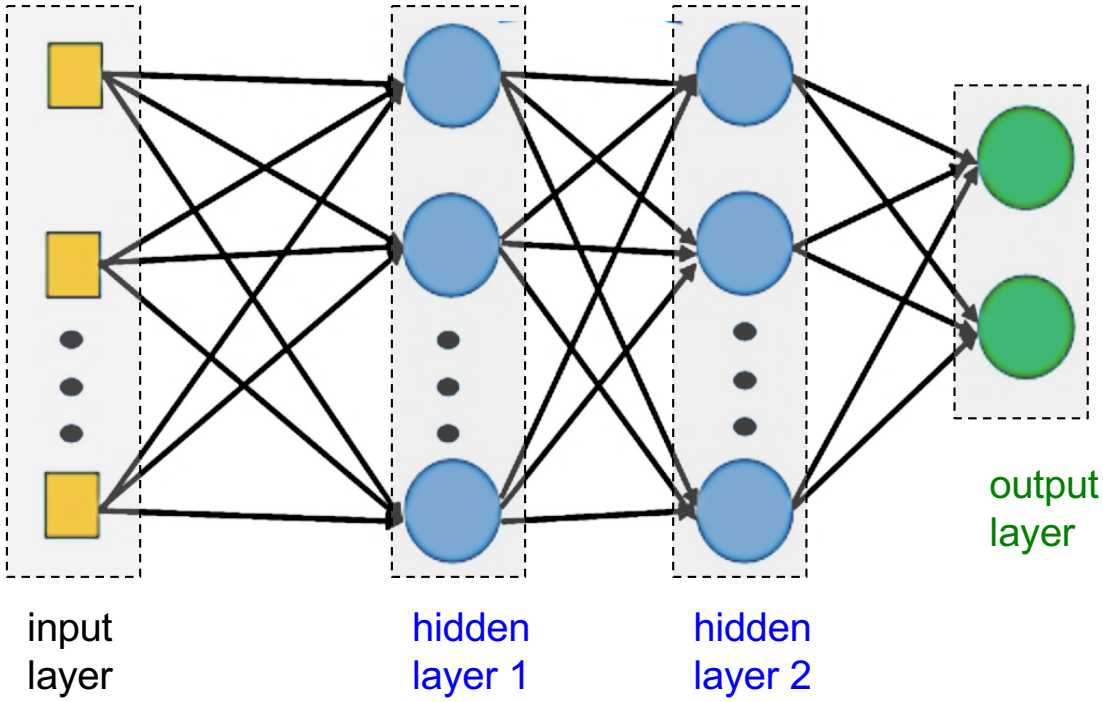
- The presence of hidden layers allows to formulate more complex functions.





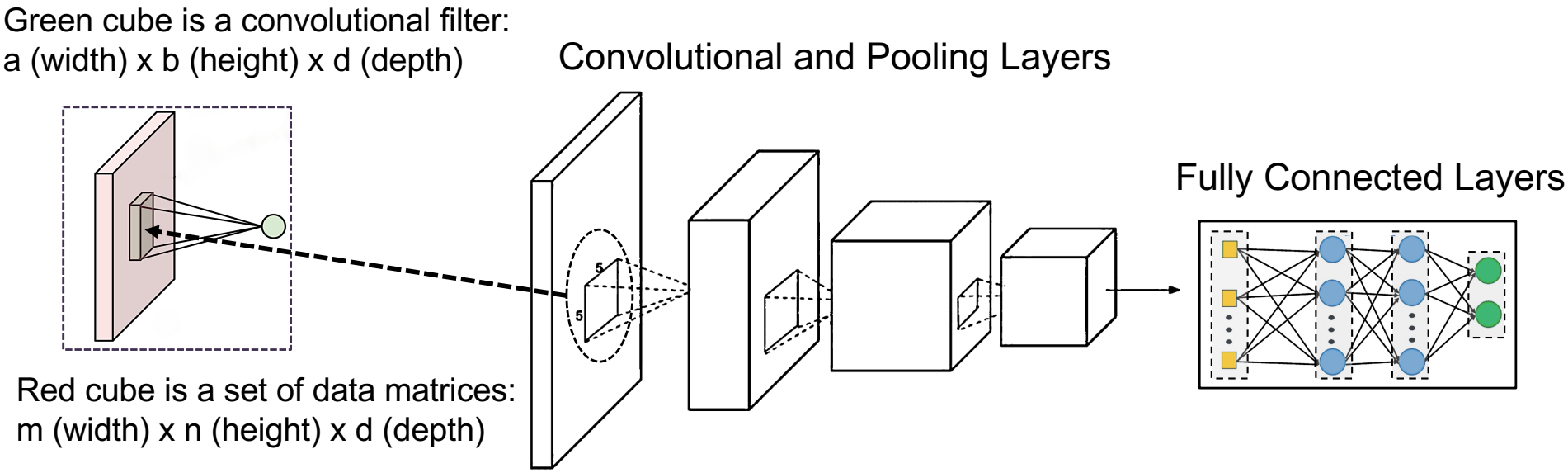
Multilayer Perceptron

- A **multilayer perceptron** (MLP), also called **feedforward artificial neural network**, consists of at least three layers of nodes: input, hidden (at least one) and output layers.
- In a **fully connected** neural network, each neuron is connected to all the neurons in the previous layer through non-zero weight.



This is a powerful tool that can help you to create a complex multi-input and multi-output function.

Convolutional Neural Network (CNN)



Main CNN recipe:

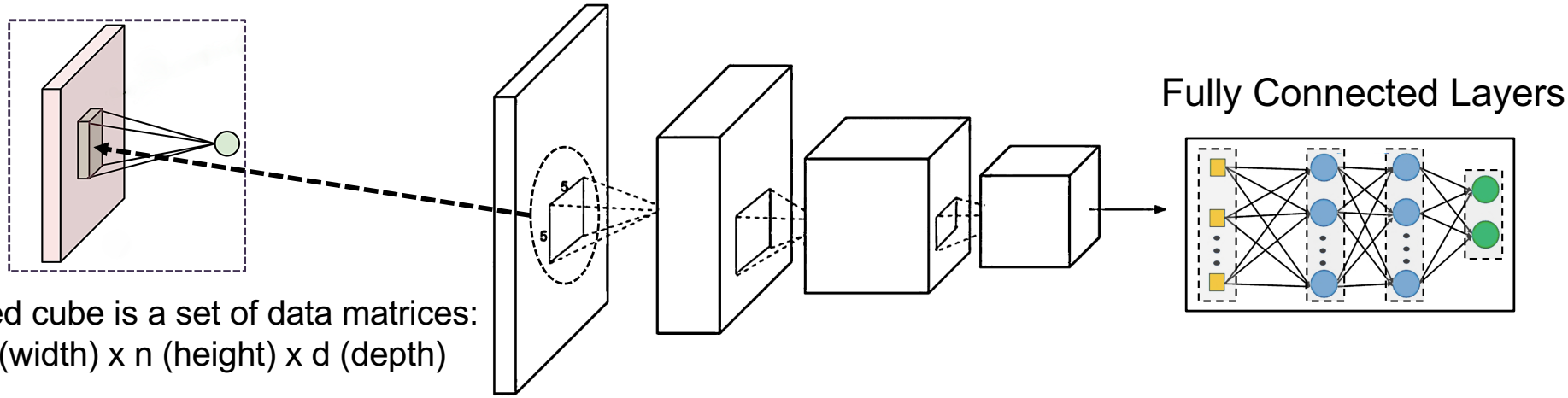
- **2D/3D** Neurons: Neurons are arranged in 2 dimensions (width and height) or 3 dimensions (width, height and depth).
- Convolutional and pooling layers: local connection, weight sharing, down-sampling.
- Fully connected layers in the end.

Convolutional Neural Network (CNN)

Green cube is a convolutional filter:
a (width) x b (height) x d (depth)

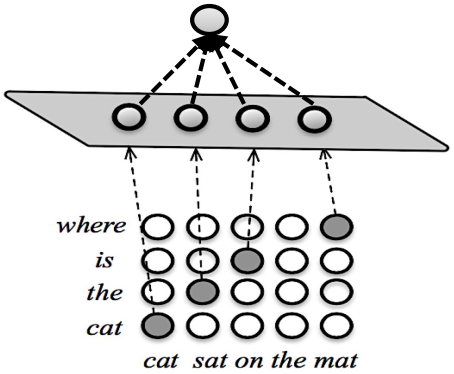
Convolutional and Pooling Layers

Fully Connected Layers

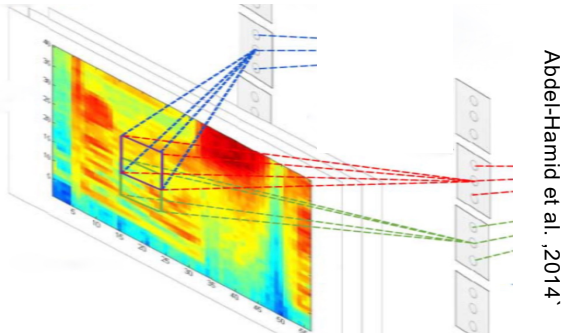


Red cube is a set of data matrices:
m (width) x n (height) x d (depth)

CNNs are good at capturing data patterns existing in a **3-D data cube** or **a 2-D data matrix**, widely used for processing images, videos, text and signals.

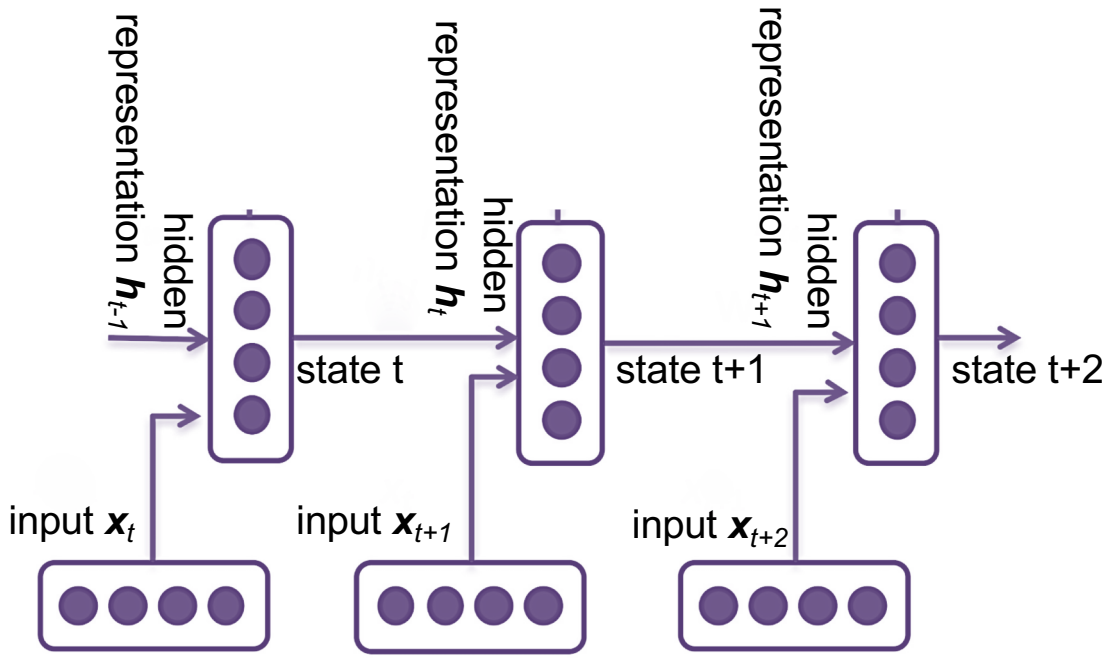


word pair similarity distribution



Energy pattern of audio signals over different frequency bands

Recurrent Neural Network (RNN)



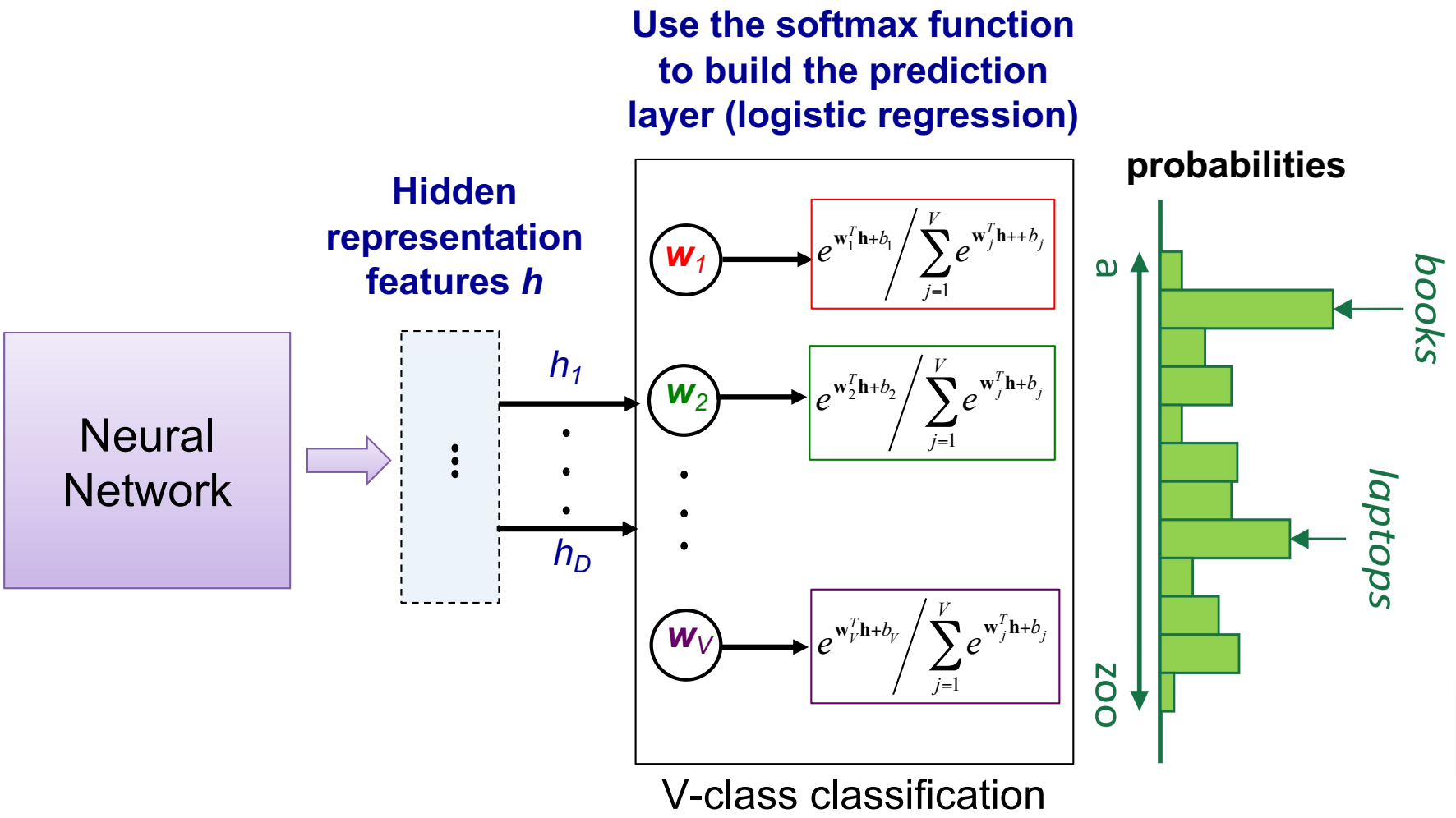
RNNs are good at capturing data patterns and encoding information in sequential data.

Prediction Layer

- Most NLP tasks can be decomposed into a series of classification tasks.
- The hidden representation vectors returned by neural networks are usually used as the input of a logistic regression model (a classifier). This is referred to as the prediction layer.
- Training neural network weights aims at minimising a classification error loss calculated using your training data.
- Cross entropy loss is a common option.

Prediction Layer

- Here is an example of the prediction layer.



Thank you!