Applied Text Analytics & Natural Language Processing

with Dr. Mahdi Roozbahani
& Wafa Louhichi

Deep Learning
Recurrent Neural Networks (RNN) – Part 1

Some of the slides are based on Ming Li (University of Waterloo – Deep Learning Part)
with some modifications

Learning Objectives

In this lesson, you will learn a deep learning model called RNN

- Recurrent Neural Networks (Sequence modeling)
- Why RNN

Name Entity Recognition (NER) using RNN

This is our document: ***Mahdi and Wafa teach NLP***

Our simple NER model will detect whether each word is referring to a person or not. In our example **1** refers to a person and **0**, not a person:

Mahdi	***and***	***Wafa***	***teach***	***NLP***
1	**0**	**1**	**0**	**0**

For NER, the location of each word in a sentence is very important

Encode Each Word into a Vector

This is our document: ***Mahdi and Wafa teach NLP***

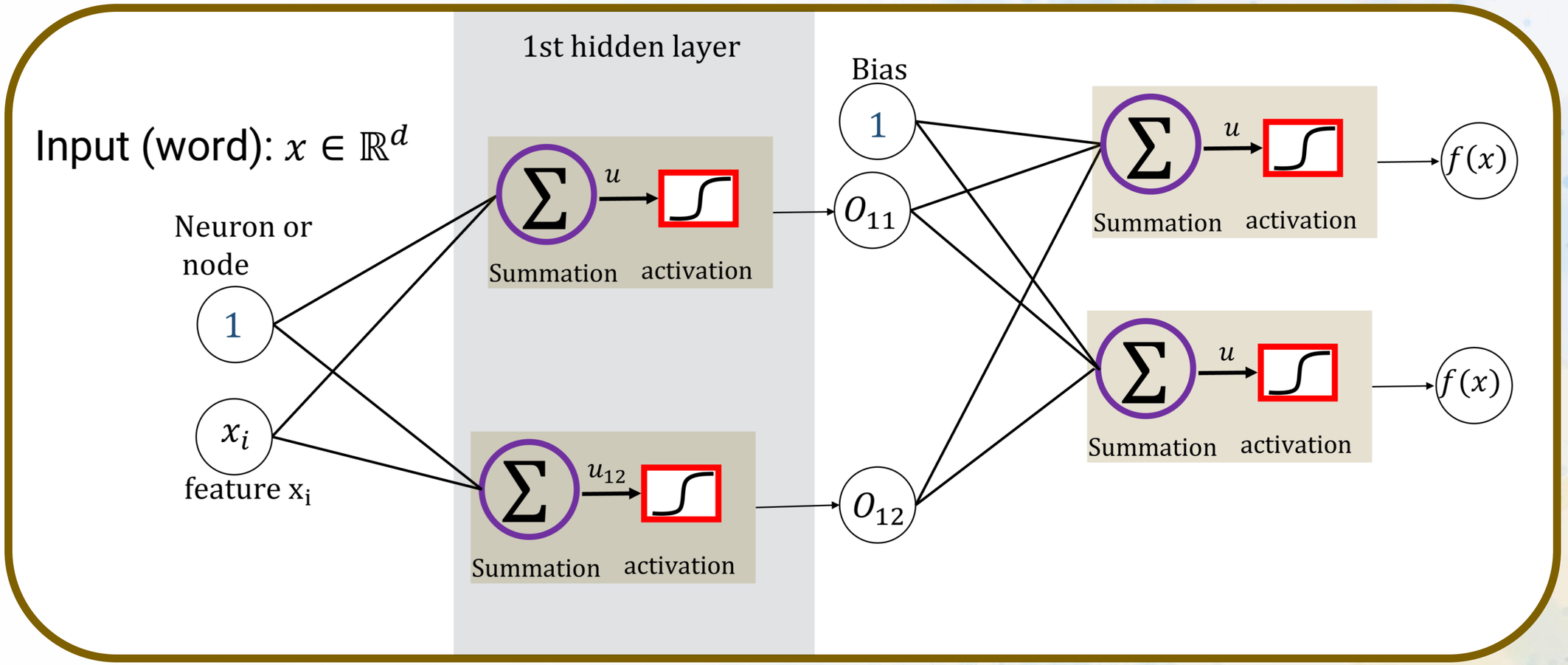
We can use an encoding method to convert work into a vector (Word2Vec, GloVe, One-hot encoding, ..)

Mahdi
And
Wafa
Teach
NLP

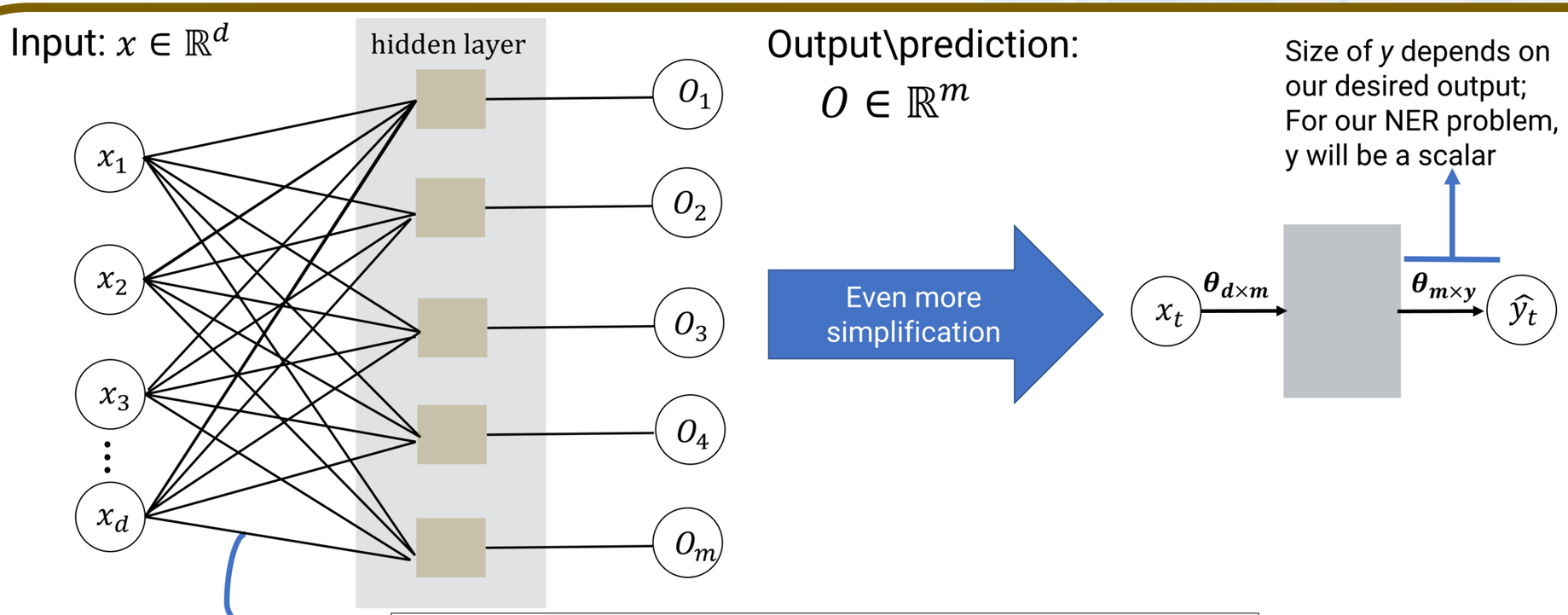
Each word (x) will be a d *dimensional* vector

$$x \in \mathbb{R}^d$$

Let's Quickly Go Over Feed Forward Networks such as ANN and CNN

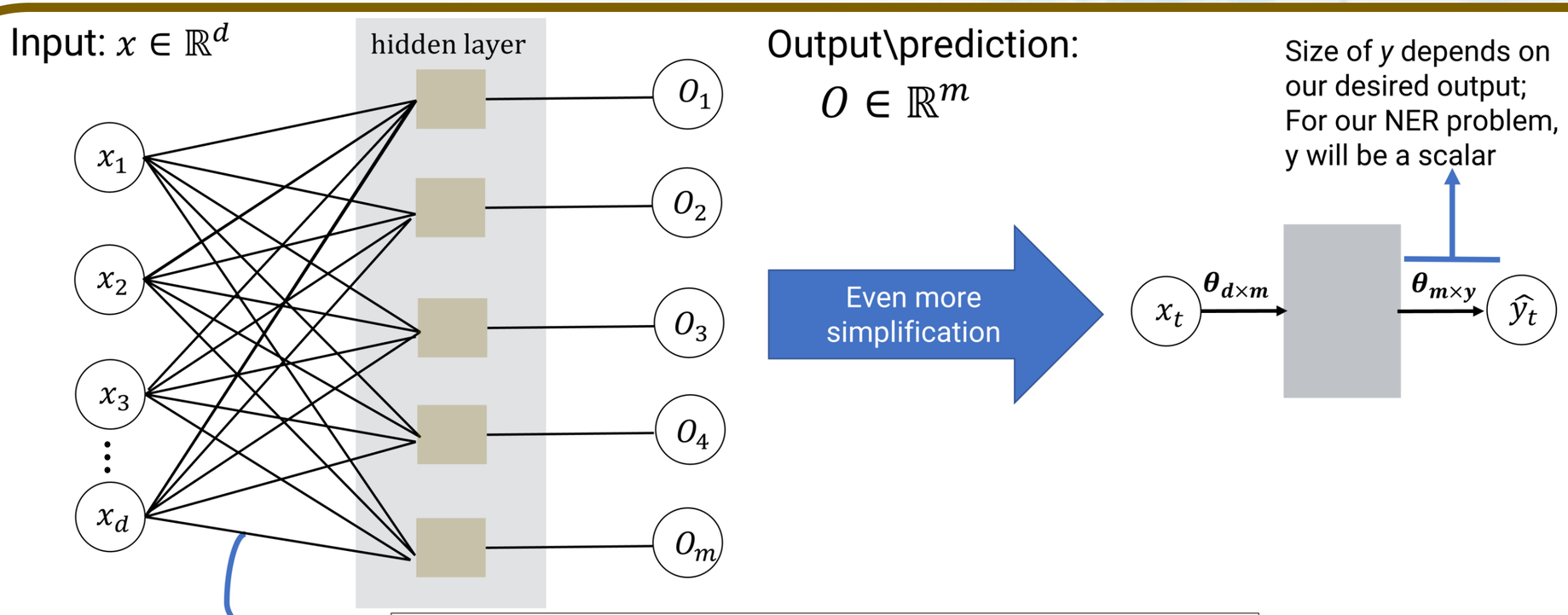


Let’s Simplify this Network a bit



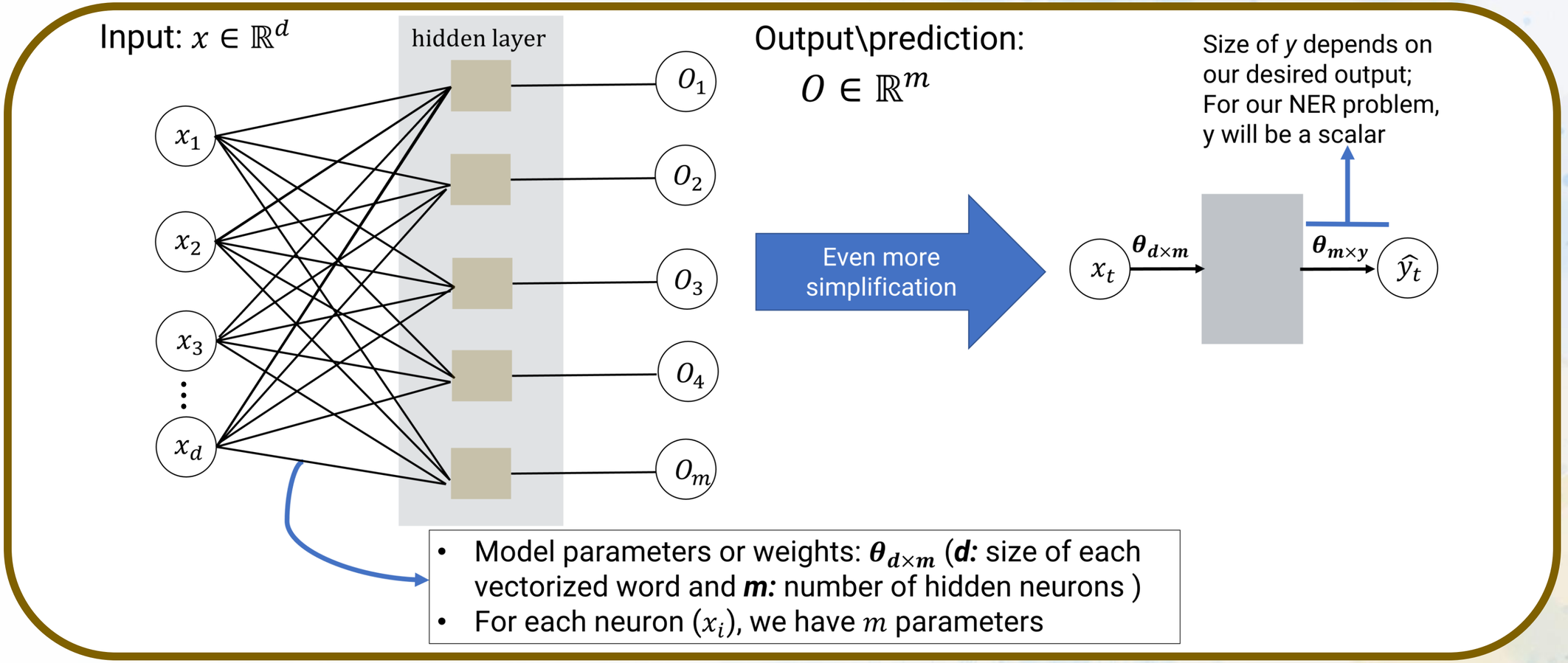
- Model parameters or weights: $\boldsymbol{\theta_{d\times m}}$ (***d:*** size of each vectorized word and ***m:*** number of hidden neurons)
- For each neuron (x_i), we have m parameters

Let’s Simplify this Network a bit



- Model parameters or weights: $\boldsymbol{\theta_{d\times m}}$ (***d:*** size of each vectorized word and ***m:*** number of hidden neurons)
- For each neuron (x_i), we have m parameters

Let’s Simplify this Network a bit



- Model parameters or weights: $\boldsymbol{\theta_{d\times m}}$ (***d:*** size of each vectorized word and ***m:*** number of hidden neurons)
- For each neuron (x_i), we have m parameters