Recurrent Neural Network

Dinesh Kumar Vishwakarma, Ph.D.

PROFESSOR, DEPARTMENT OF INFORMATION TECHNOLOGY

DELHI TECHNOLOGICAL UNIVERSITY, DELHI.

Webpage: http://www.dtu.ac.in/Web/Departments/InformationTechnology/faculty/dkvishwakarma.php

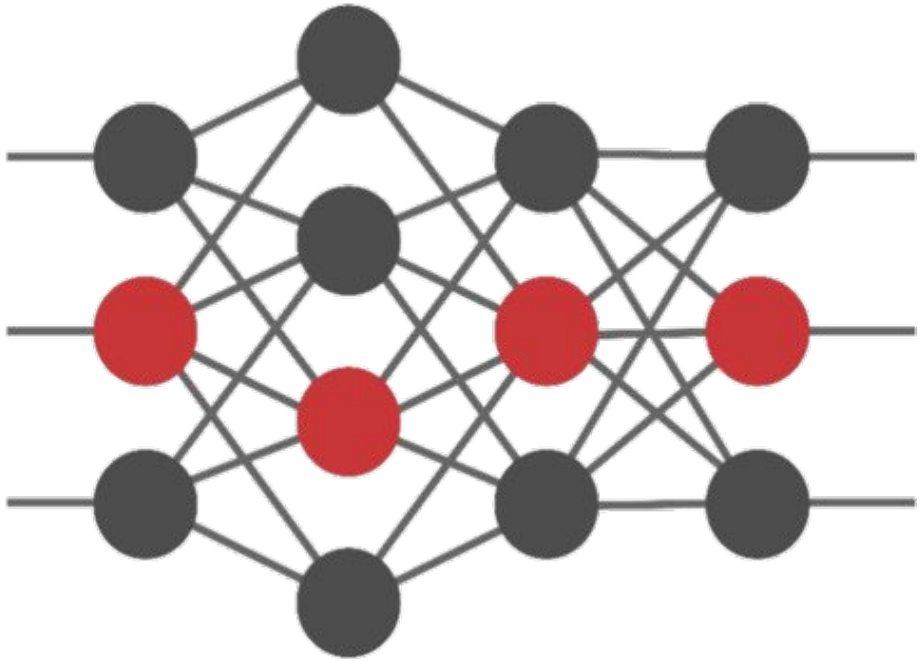
Email: dinesh@dtu.ac.in

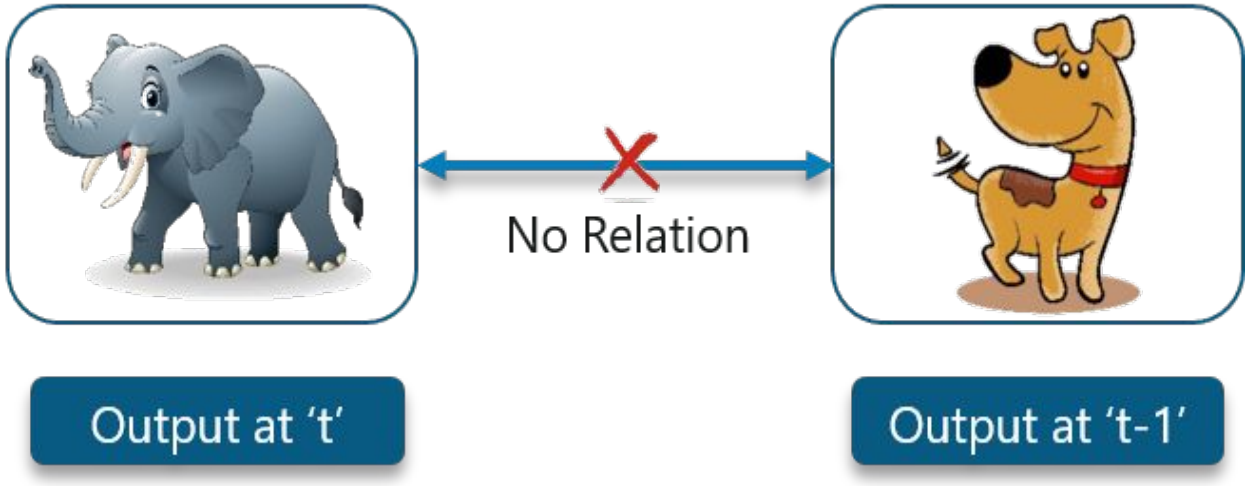
Outline

- Why Not Feed-forward Networks?
- What Are Recurrent Neural Networks?
- How To Train Recurrent Neural Networks?
- Vanishing And Exploding Gradients
- Long Short Term Memory (LSTM) Networks
- LSTM Use-Case

Why Not Feed-forward Networks?

- Consider an **image classification,** where NN **trained** to **classify images** of various **animals** such as a **cat** or a **dog,** the NN provides an **output** with a **corresponding label** to the image of a cat or a dog **respectively.**



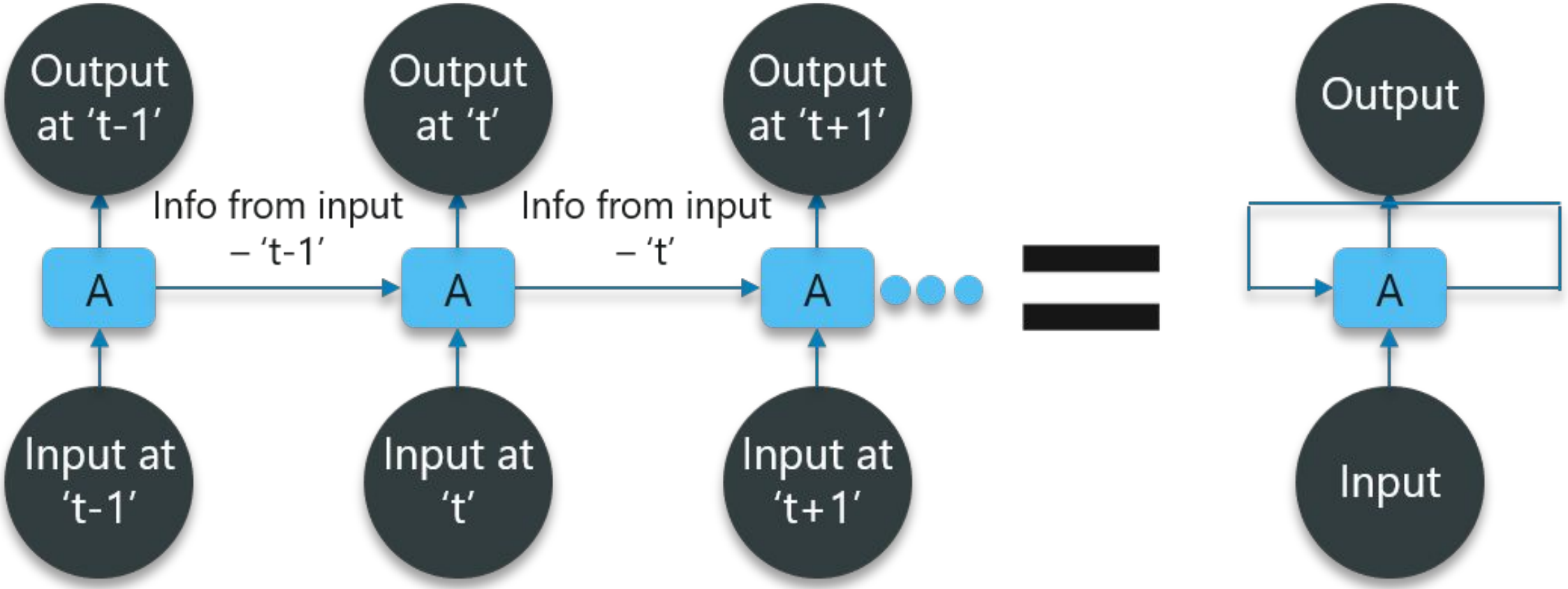


Why Not Feed-forward Networks?...

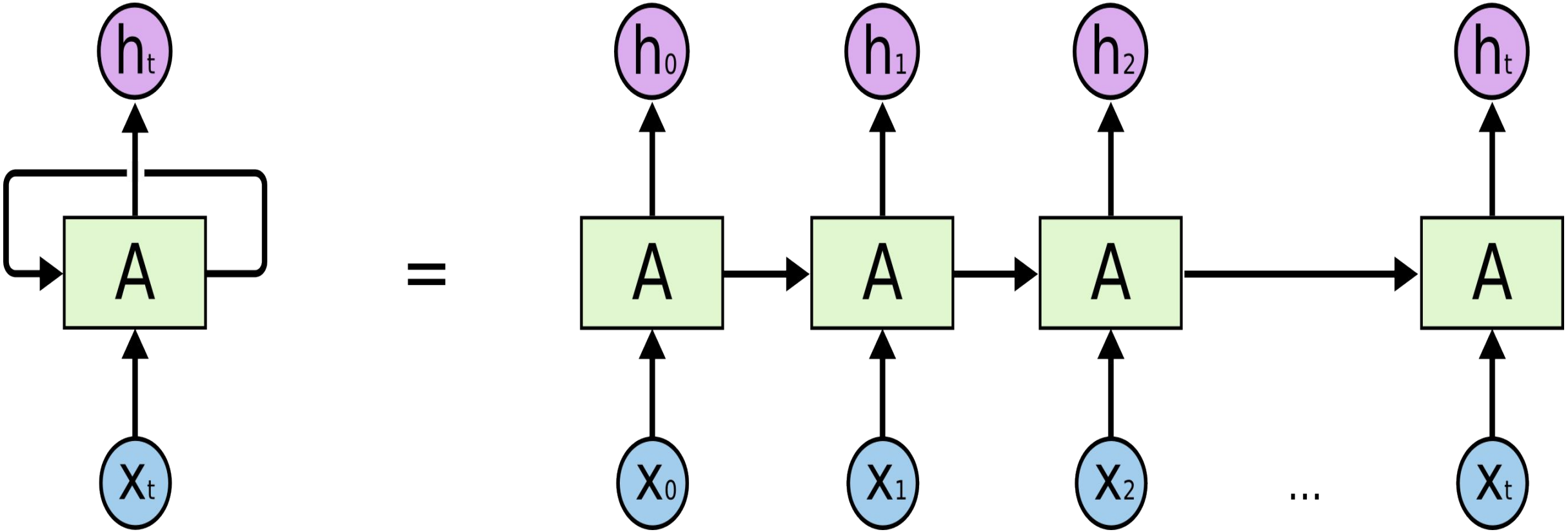
- Consider a real case of **reading** a **book,** where with **every page** you move forward into, you need the **understanding** of the **previous pages** to make **complete** sense of the **information.**
- With a **feed-forward network** the **new** output at time $'t+1'$ has **no relation** with outputs at either time $t, t-1, or\ t-2$.
- So, feed-forward networks **cannot be used** when **predicting** a **word** in a sentence as it will have no **absolute relation** with the **previous** set of **words.** But, with **Recurrent Neural Networks,** this challenge can be **overcome.**
- A **recurrent neural network** (**RNN**) is a class of artificial neural networks where connections between nodes form a directed graph along a **temporal sequence**

Why Not Feed-forward Networks?...

- Consider a diagram: RNN



Generalized RNN

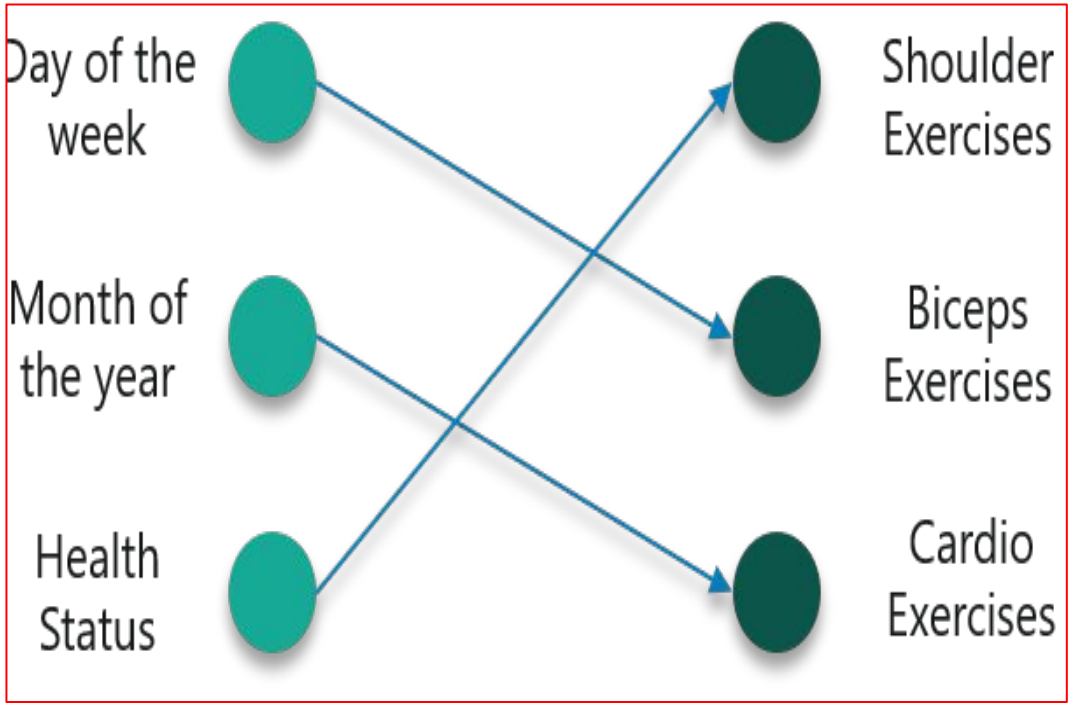


Recurrent Neural Network?

- Recurrent Networks are a **type** of **artificial neural network** designed to **recognize patterns** in sequences of data, such as **text,** genomes, handwriting, the spoken word, numerical times **series** data emanating from sensors, **stock markets** and **government agencies.**
- Consider an Example of **gym,** where **trainer** has given a **schedule** for workout:

Recurrent Neural Network?...

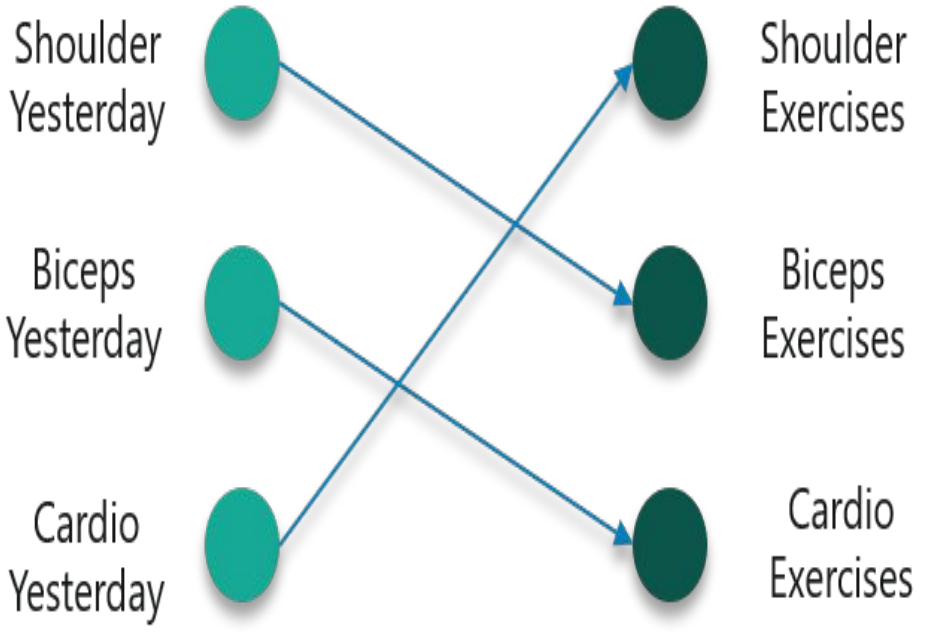
- These exercise are repeated every week in proper order.
- Let we try to predict the exercise of day using Feedforward NN.
- The inputs are **day, month** and the **health status.** A neural network has to be **trained** using these inputs to provide us the with the **prediction** of the **exercises.**



This is not appropriate to do

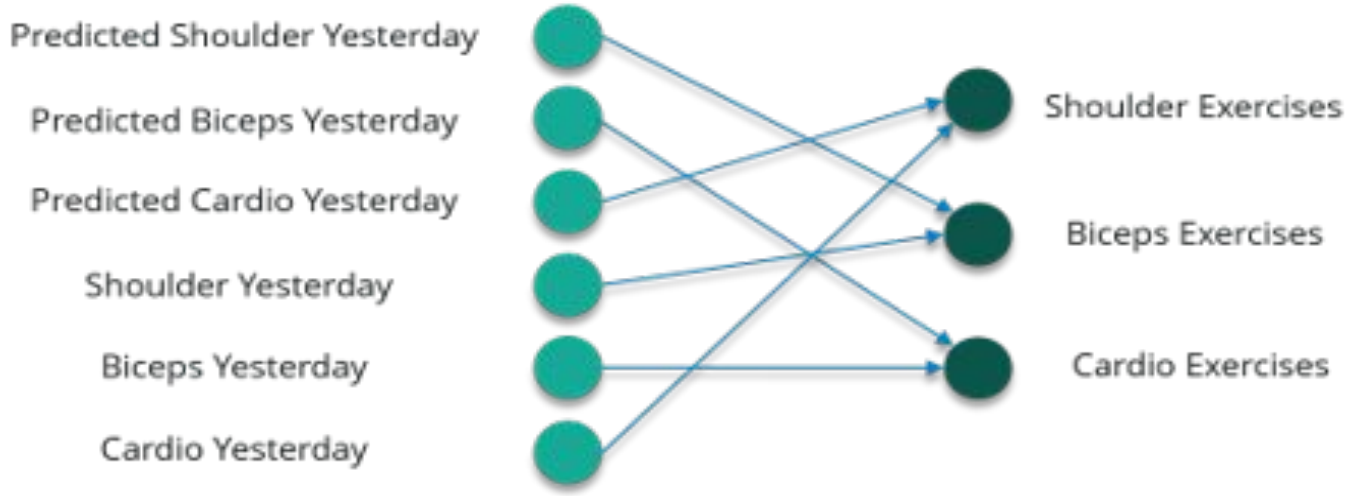
Recurrent Neural Network?...

- To **fix this,** we can make use of the concept of **Recurrent Neural Networks** can be buildup.
- Consider the **inputs** to be the **workout** done on the **previous day.**
- So if a person did a **shoulder workout** yesterday, he can do a **bicep exercise** today and this **goes on** for the **rest** of the **week** as well.



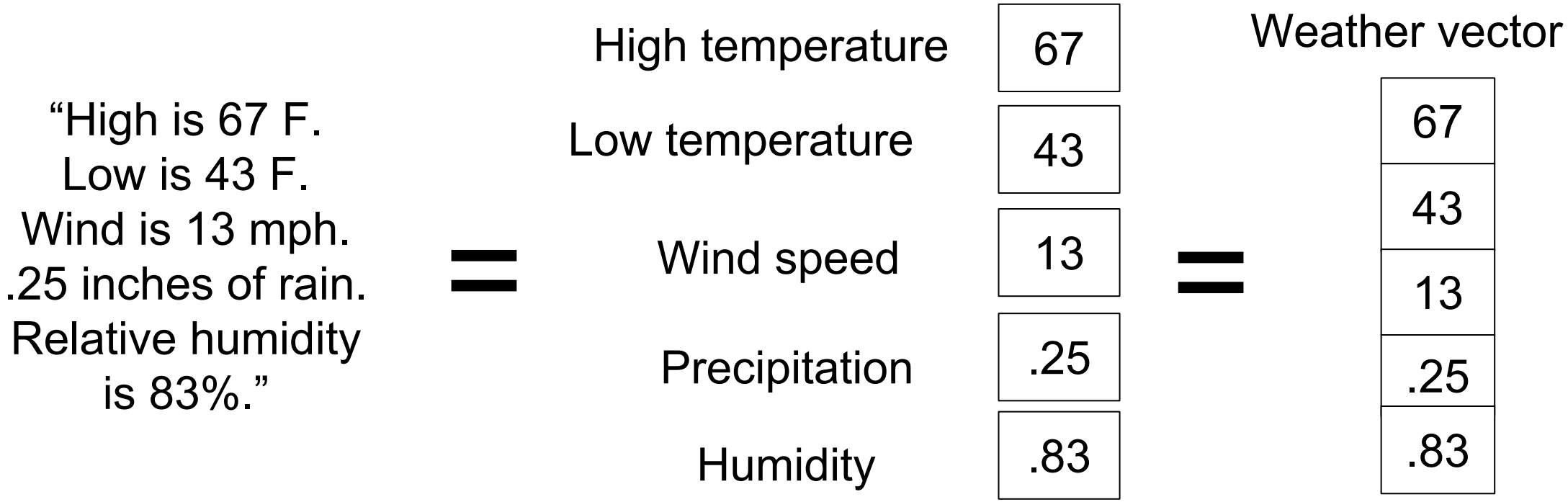
Recurrent Neural Network?...

- However, if you happen to **miss a day** at the gym, the data from the **previously attended timestamp** can be **considered** as shown below.
- If a **model** is trained based on the data it can **obtain** from the **previous exercises,** the output from the model will be **extremely accurate.**



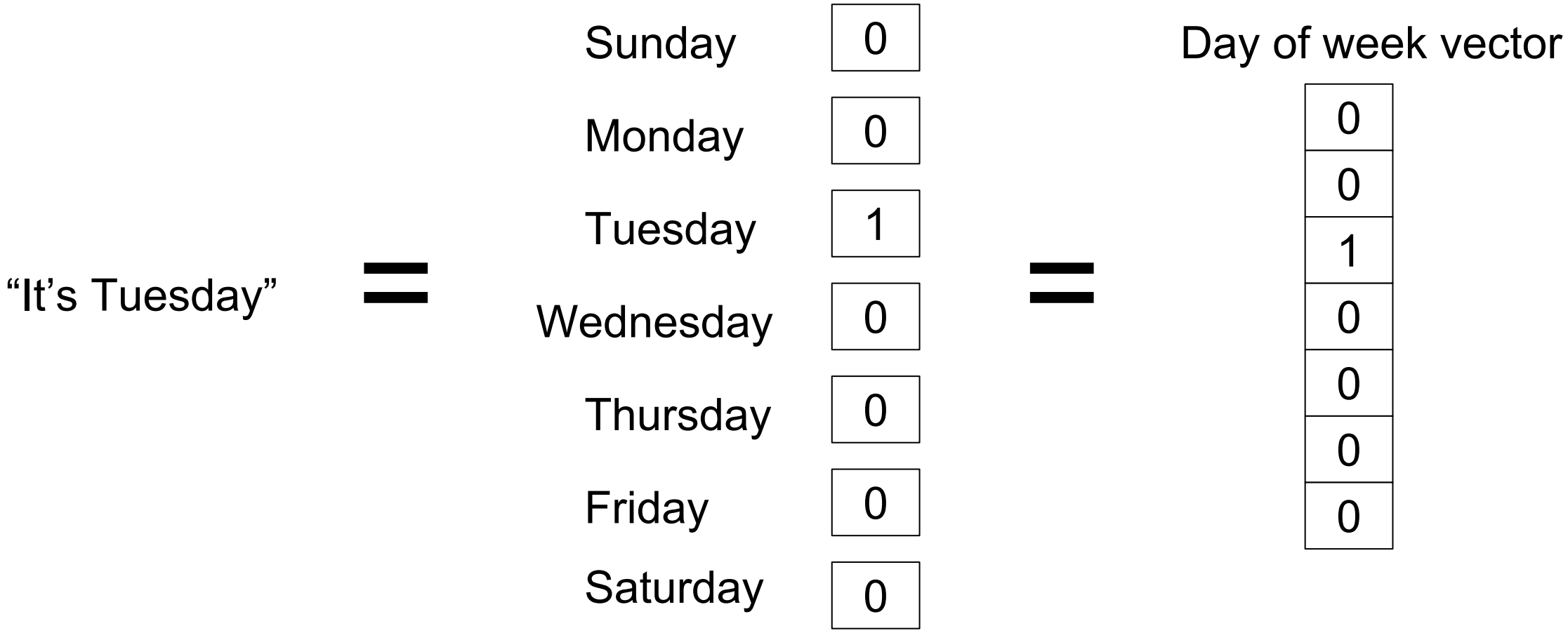
A Data as a Vector

- **Vectors** are **numbers** which are input to the **model** to **denote** if you have **done** the **exercise** or **not.**
- A vector is a list of values: *Example 1*



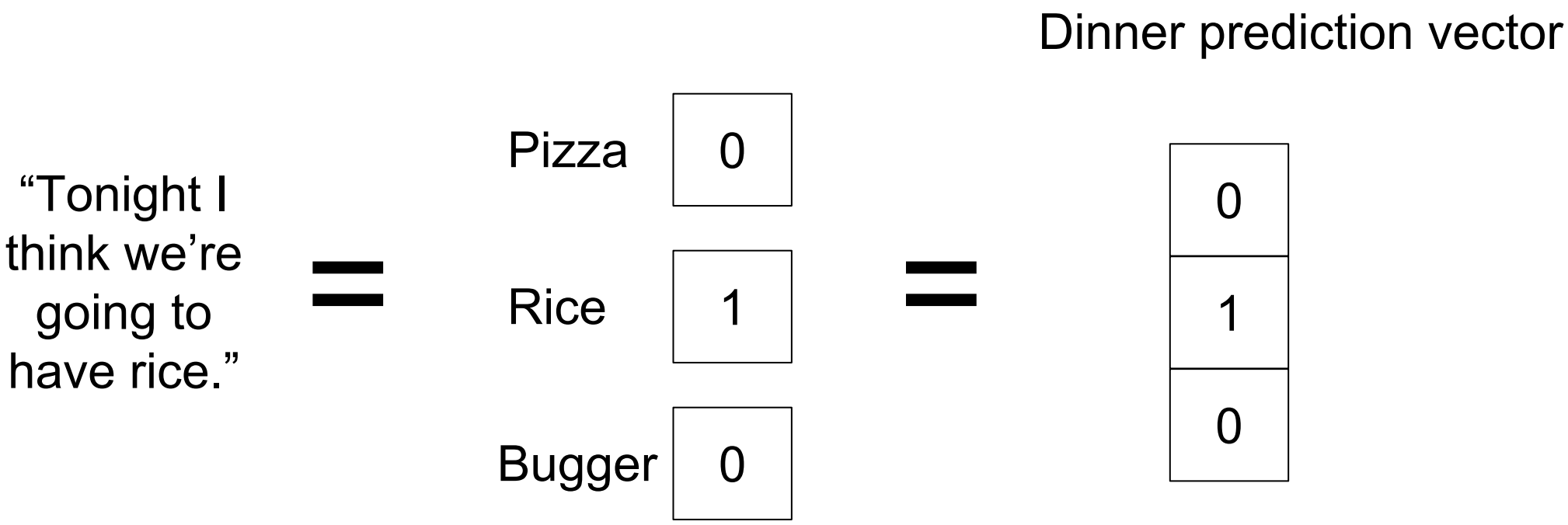
A Data as a Vector…

- A vector is a list of values: *Example 2*



A Data as a Vector…

- A vector is a list of values: *Example 3*

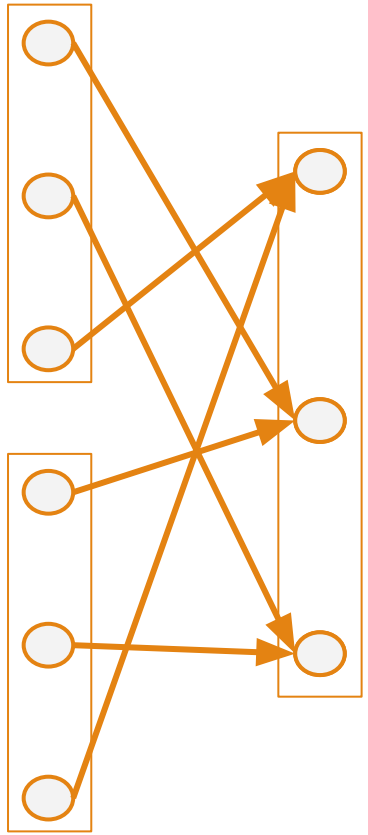


Example

- Model is more appropriate, which gives prediction dinner based on previous day dinner

Prediction of Yesterday
- predicted Pizza for yesterday
- predicted Rice for yesterday
- predicted Bugger for yesterday

Dinner Yesterday
- Pizza yesterday
- Rice yesterday
- Bugger yesterday

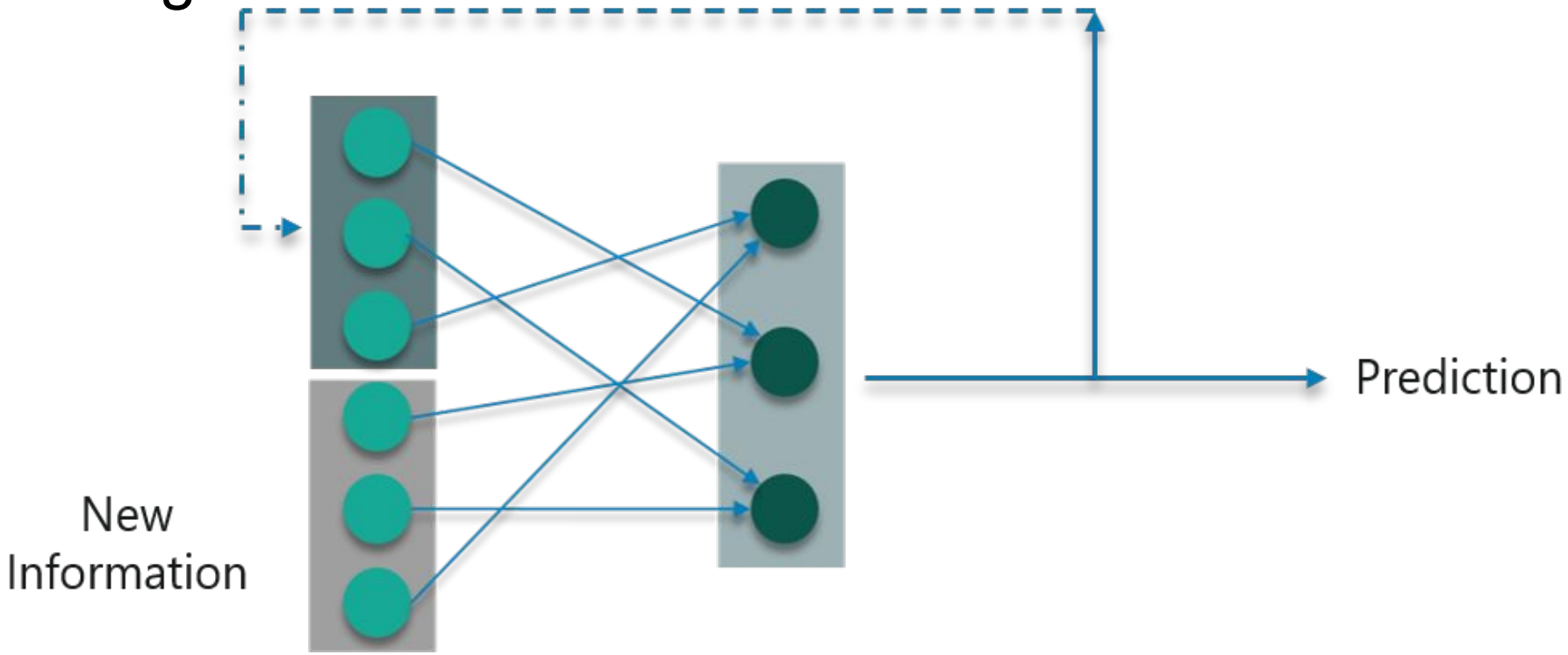


Dinner of Today
- Pizza 
- Rice 
- Bugger 

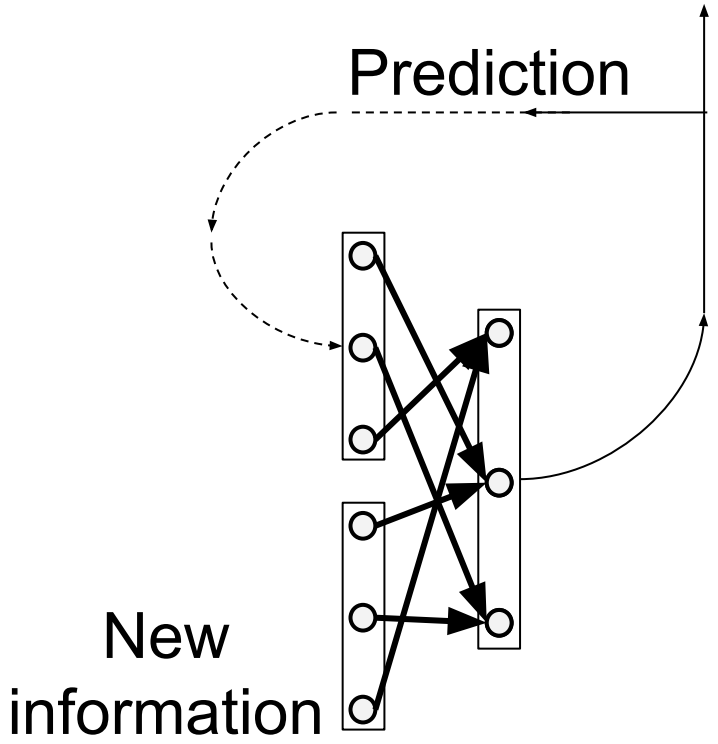
It is kind of RNN

Model of these Examples

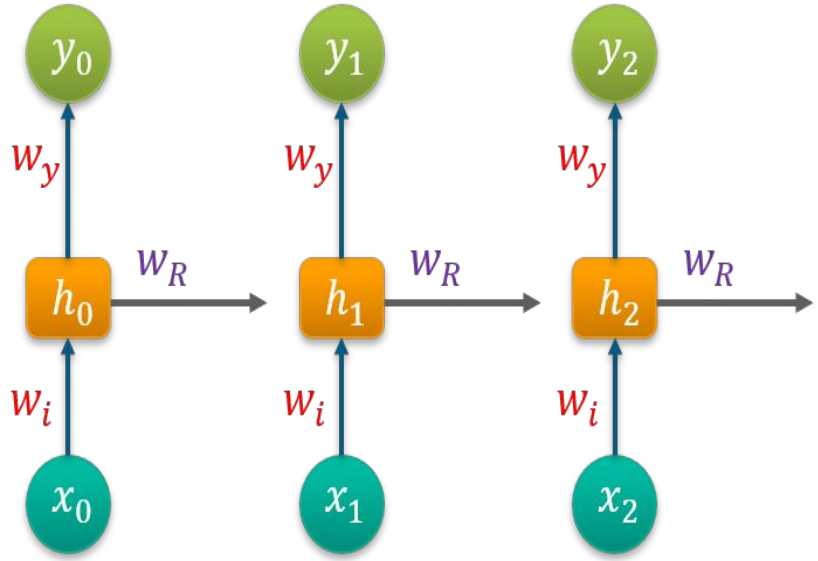
- These are all examples, a correct prediction can be made based upon previous output or state. Hence, a model can be buildup as shown in below Fig.



Model: Recurrent Neural Network



- A mathematical model can be buildup, where **'w'** to be the **weight matrix** and **'b'** being the **bias.**
- At time **t=0,** input is **'x0'** and the task is to figure out what is **'h0'.** Putting **t=0** in the **equation** and obtaining the function **h(t) value.** Next, the value of **'y0'** is found out using the **previously calculated values** when applied to the **new formula**



$$h^{(t)} = g_h\left(w_i x^{(t)} + w_R h^{(t-1)} + b_h\right)$$

$$y^{(t)} = g_y\left(w_y h^{(t)} + b_y\right)$$

Example of NN

- Consider the preparation of food items based on weather conditions.
- Consider a NN model to do this.
- Input weather is '**sunny**' then output is '**ice-cream**'
- Input weather is '**Rainy** ' then output is '**French Fries**'

Weather

Example of NN…

- Consider these are represented as vector

$$\begin{bmatrix}1\\0\\0\end{bmatrix}$$

Ice Cream

$$\begin{bmatrix}0\\1\\0\end{bmatrix}$$

French Fries

$$\begin{bmatrix}0\\0\\1\end{bmatrix}$$

Pizza

Sunny

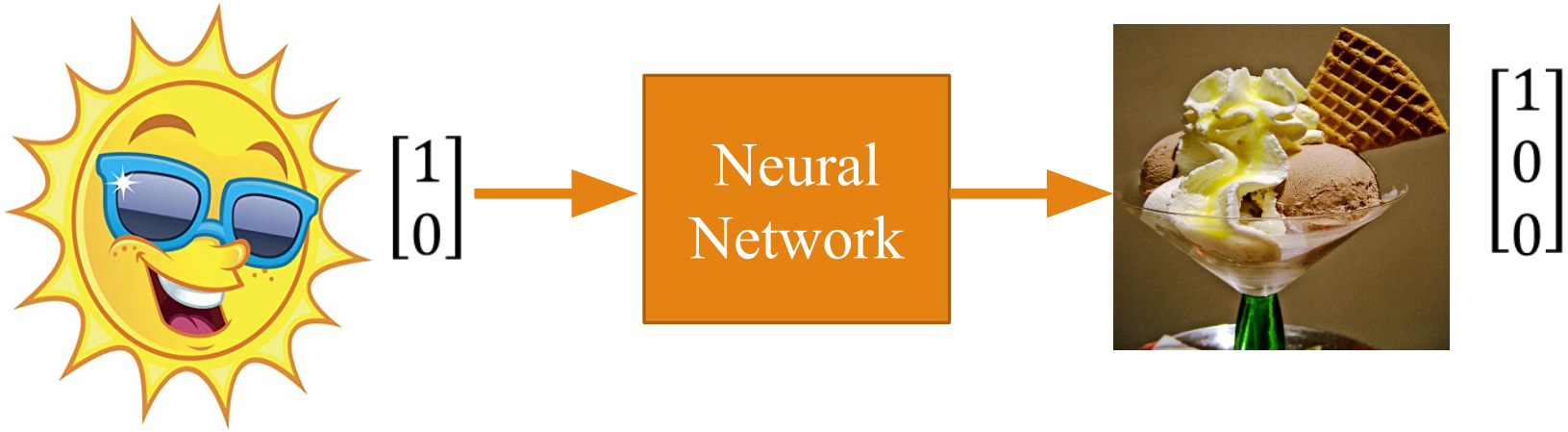
$$\begin{bmatrix}1\\0\end{bmatrix}$$

Rainy

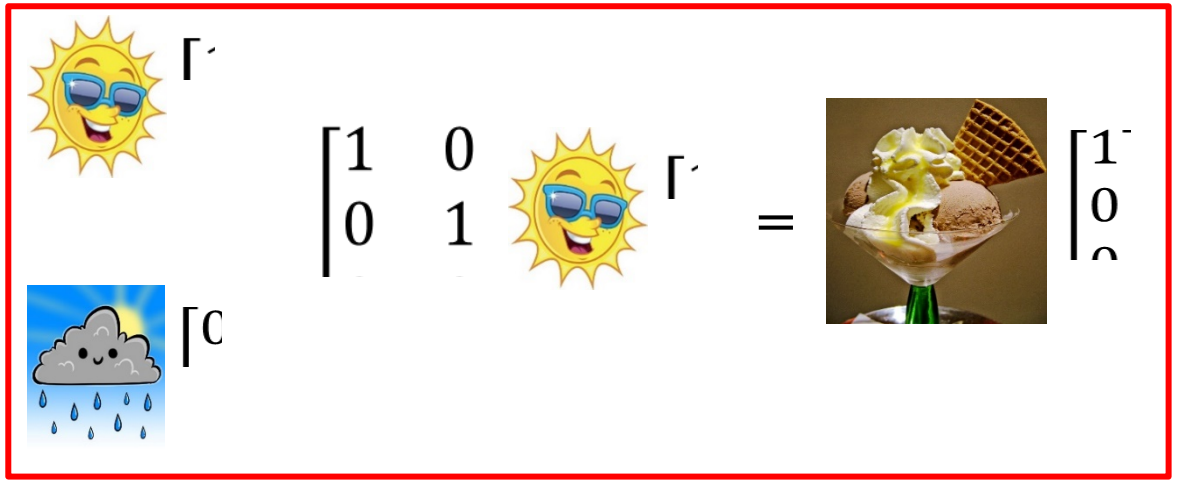
$$\begin{bmatrix}0\\1\end{bmatrix}$$

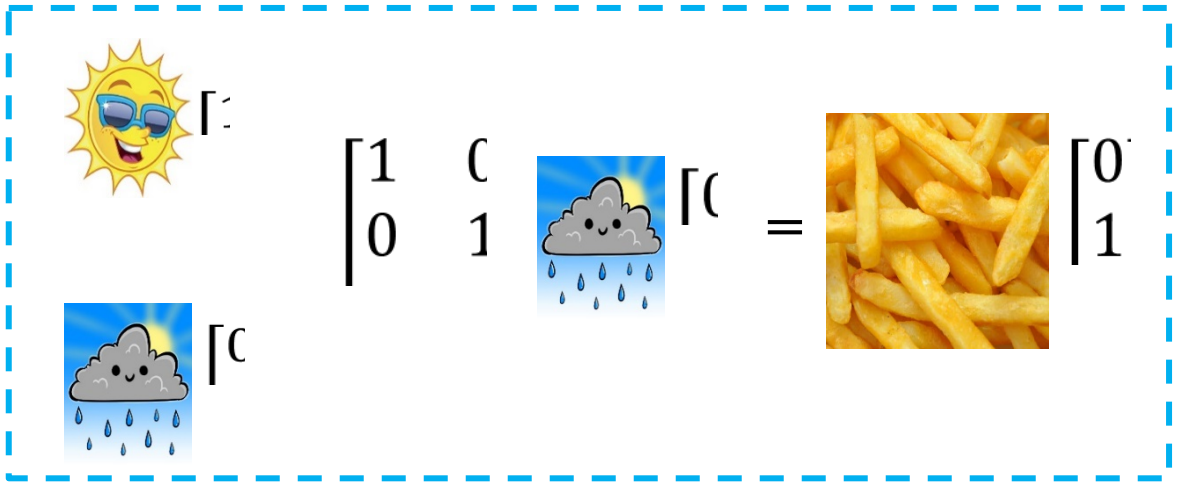
Example of NN…

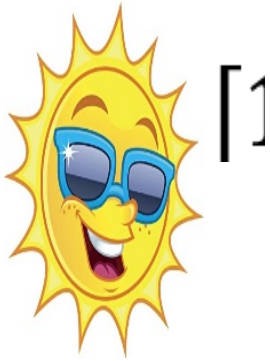
- Consider these are represented as vector

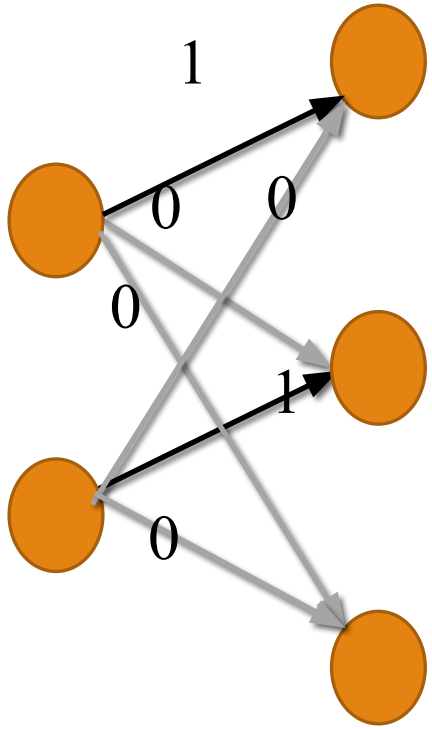


Example of NN…Representation as Vector



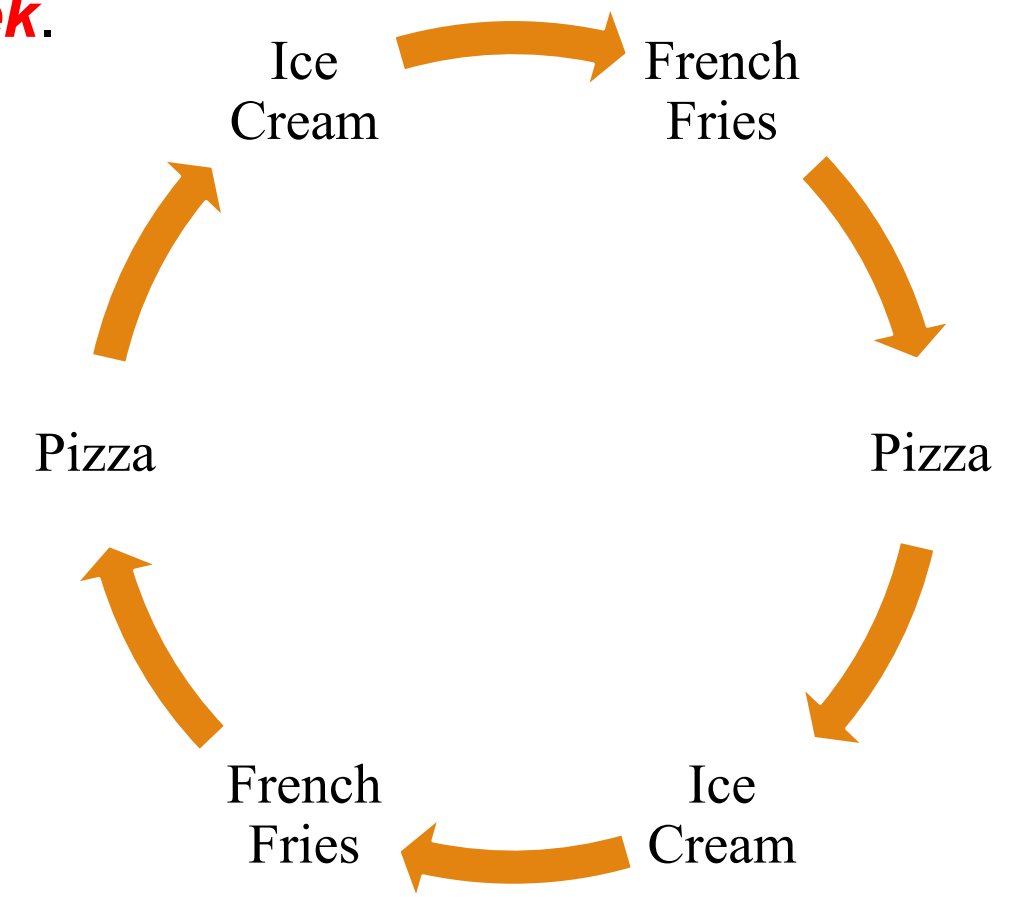




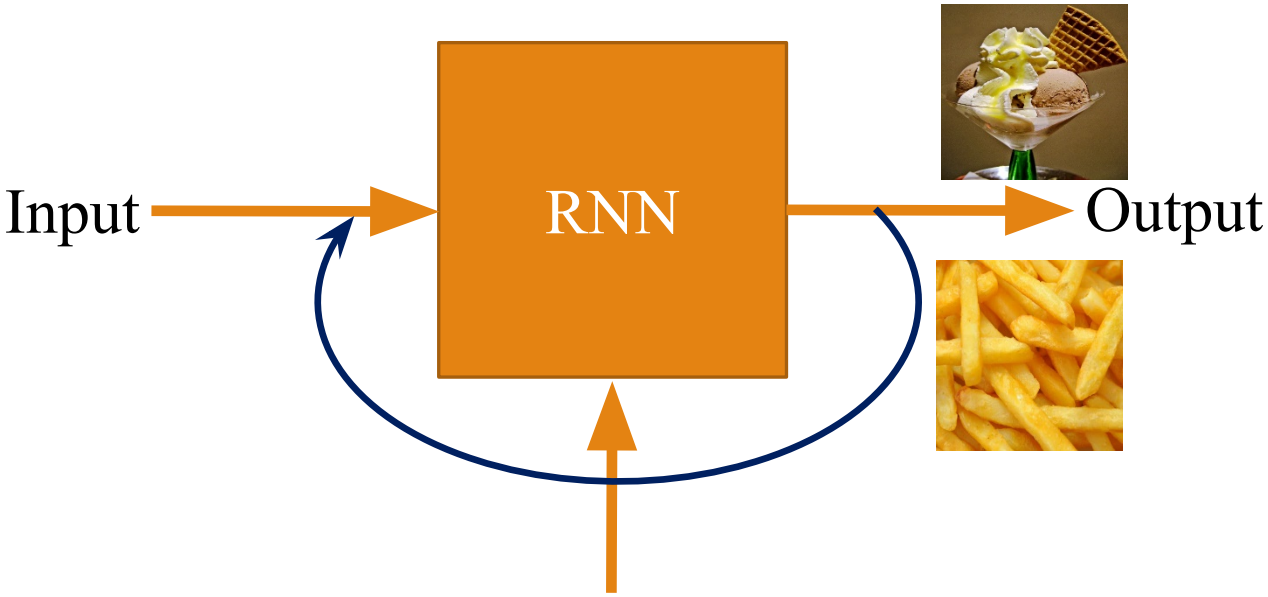


Good Example of Cooking Food

- A good cook, always have understanding what he/she has served on **previous occasions/days**. ***Day wise in a week***.



Model of Cooking Food



Simple Neural Network

$$\begin{bmatrix}1\\0\end{bmatrix}$$

$$\begin{bmatrix}1\\0\end{bmatrix}$$

$$\begin{bmatrix}0\\1\end{bmatrix}$$

$$\begin{bmatrix}0\\0\end{bmatrix}$$

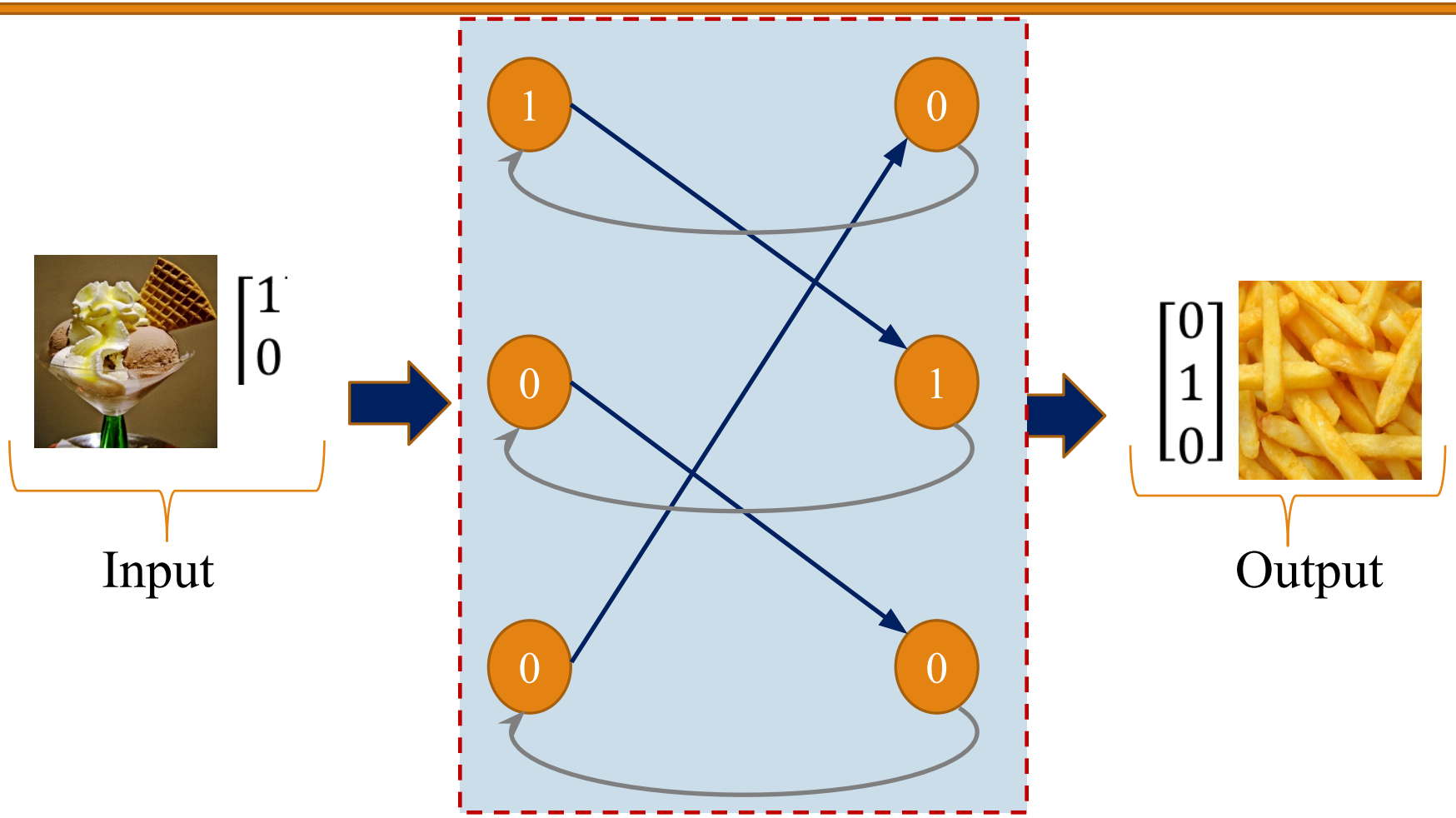
$$\begin{bmatrix}0\\1\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 1\\1 & 0 & 0\\0 & 1 & 0\end{bmatrix} =$$

$$\begin{bmatrix}0\\0\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 1\\1 & 0 & 0\\0 & 1 & 0\end{bmatrix} =$$

Recurrent Neural Network



Schedule of Food with Weather

- Consider if weather is sunny then person will out and enjoy the weather and if its rainy then person will sty at home and cooked the food as per schedule.

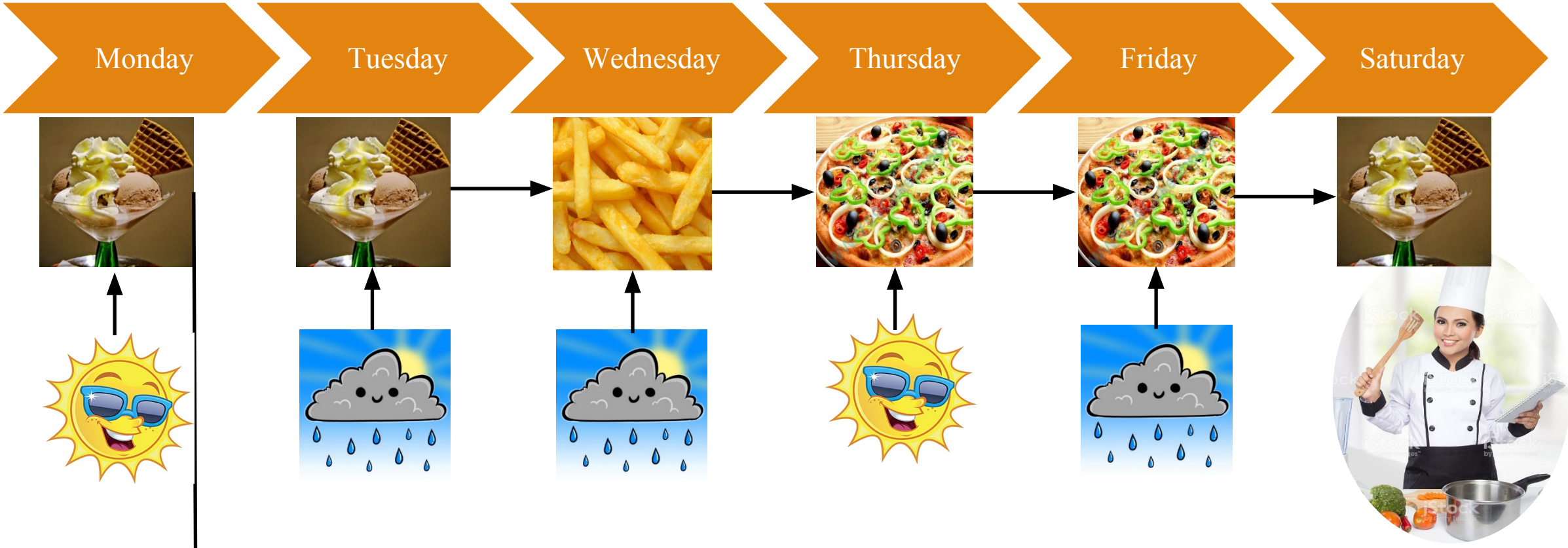
$\begin{bmatrix}1\\0\end{bmatrix}$

Same as yesterday

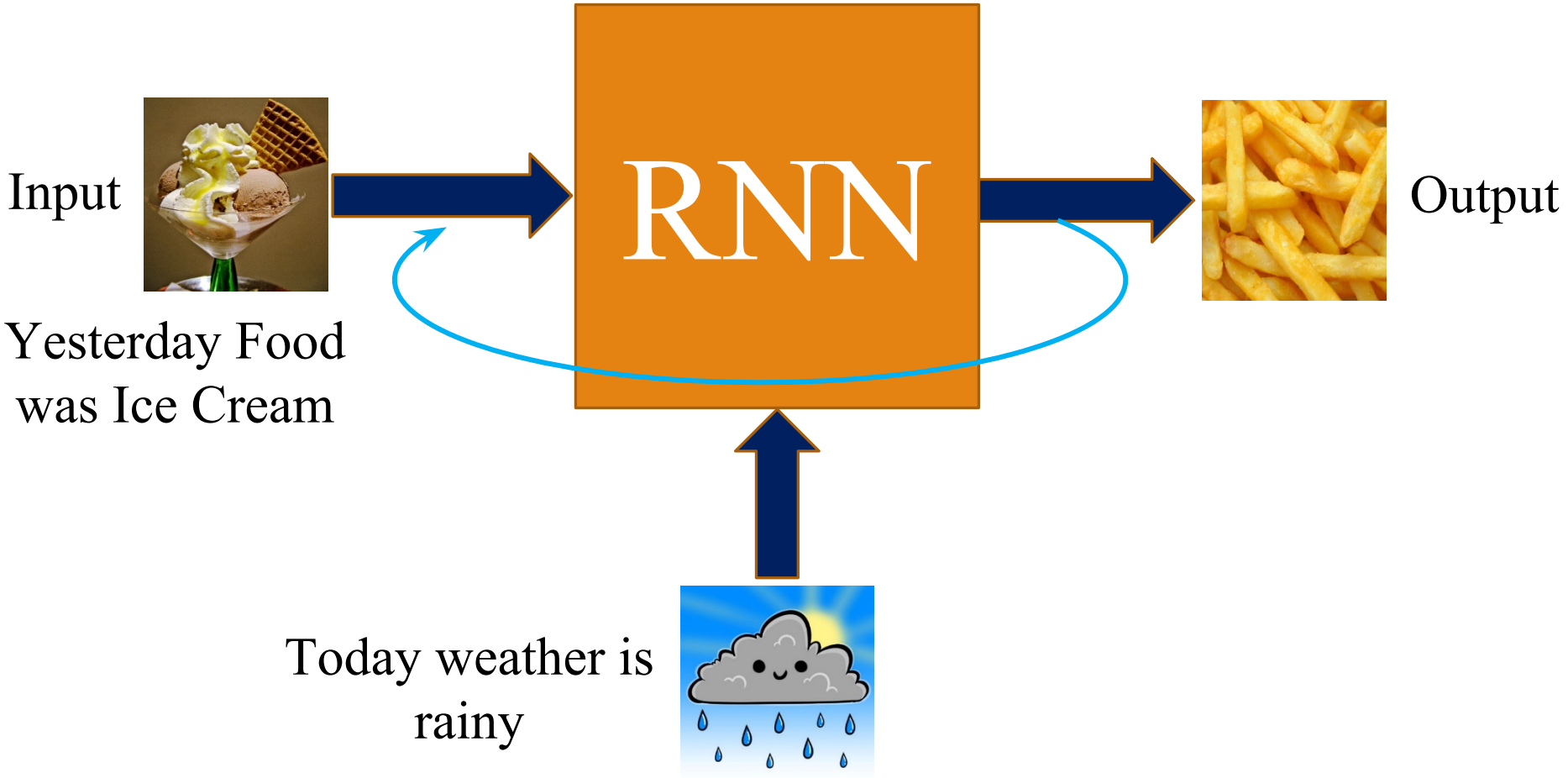
$\begin{bmatrix}0\\1\end{bmatrix}$

Next Dish

Cooking Schedule of Week



Model of Cooking Schedule : RNN



Vector Representation of Food Items

- Consider these are represented as vector

$$\begin{bmatrix}1\\0\\0\end{bmatrix}$$

Ice Cream

$$\begin{bmatrix}0\\1\\0\end{bmatrix}$$

French Fries

$$\begin{bmatrix}0\\0\\1\end{bmatrix}$$

Pizza

Sunny

$$\begin{bmatrix}1\\0\end{bmatrix}$$

Rainy

$$\begin{bmatrix}0\\1\end{bmatrix}$$

Food Matrix

- Consider these are represented as vector

$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$

Ice Cream

$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$

French Fries

$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$

Pi

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \hline 0 \\ 1 \\ 0 \end{bmatrix}$$

Same

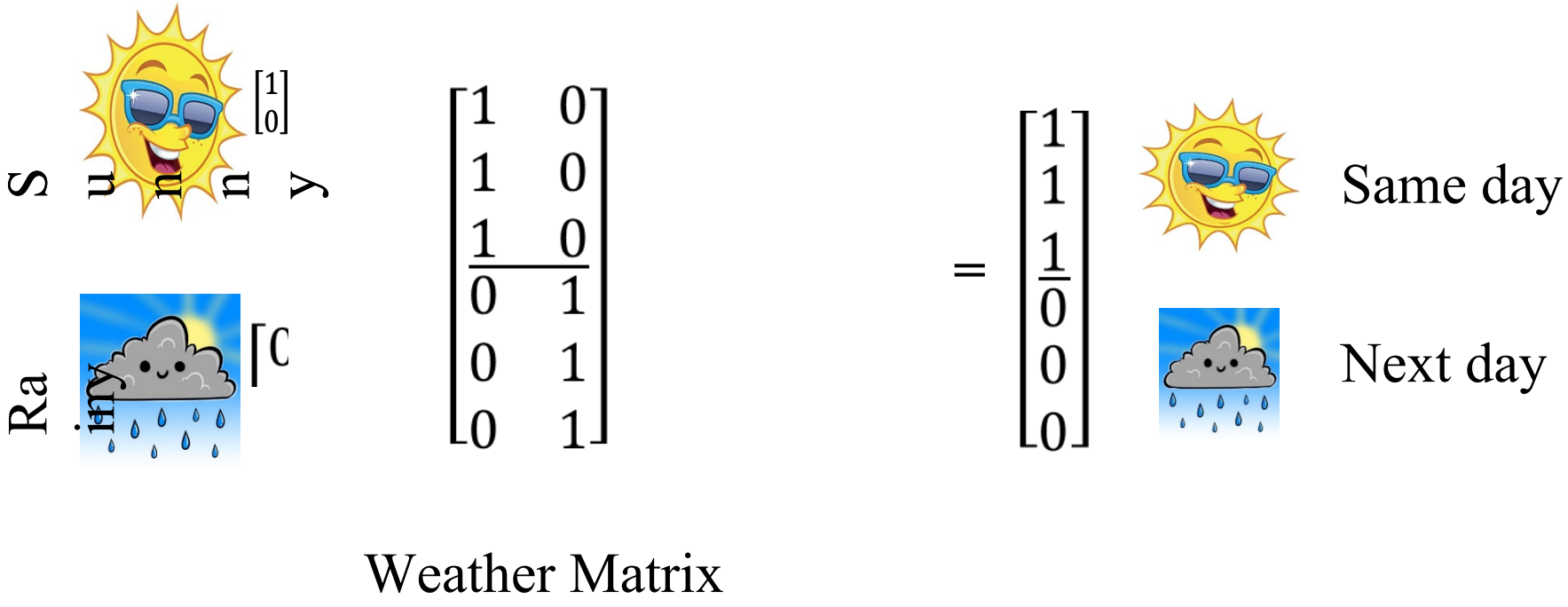
Next Day

Concenation of todays and tomorrow food

Similarly for other items too

Weather Matrix

- Consider these are represented as vector



A More Complicated RNN

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad + \quad \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ \hline 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \rightarrow$$

Food | Add | Weather | Merge

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ \hline 0 \\ 1 \\ 0 \end{bmatrix}$$  Same

 Next Day

$$+ \quad \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hline 1 \\ 1 \\ 1 \end{bmatrix}$$  Same

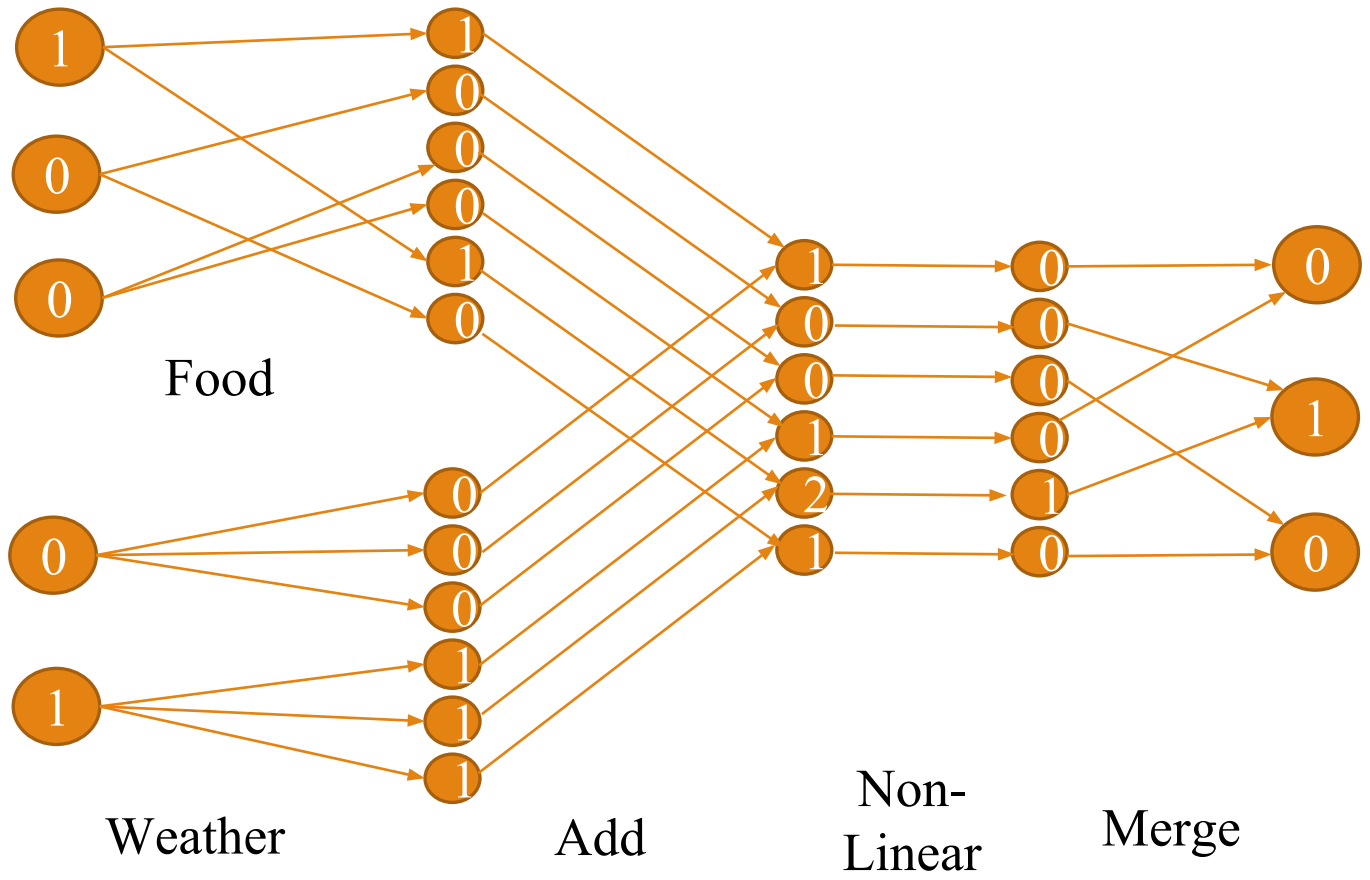
 Next Day

$$= \begin{bmatrix} 1 \\ 0 \\ 0 \\ \hline 1 \\ 2 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ \hline 1 \\ 2 \\ 1 \end{bmatrix} \xrightarrow{\text{Non Linear}} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hline 0 \\ 1 \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0+0 \\ 0+1 \\ 0+0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$ 

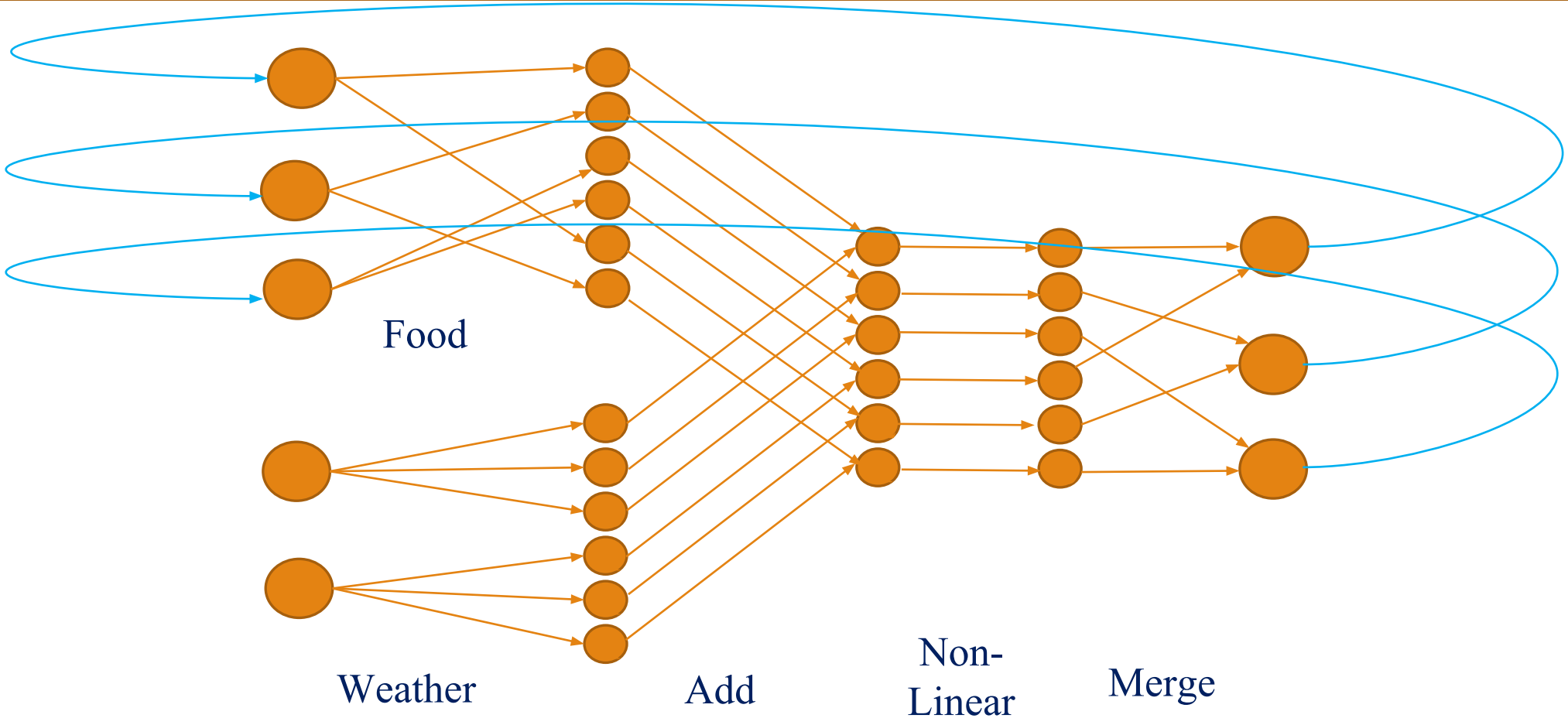
French fries, food of next day

RNN Model



Food of Next Day

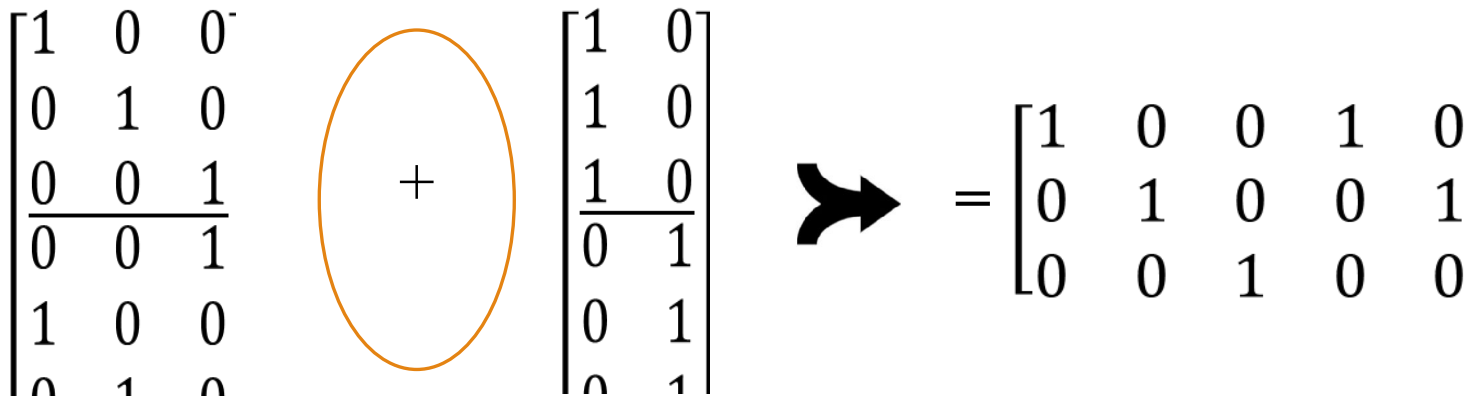
RNN Model…



Output Vector is fed to input : **Recurrent**

Training RNN

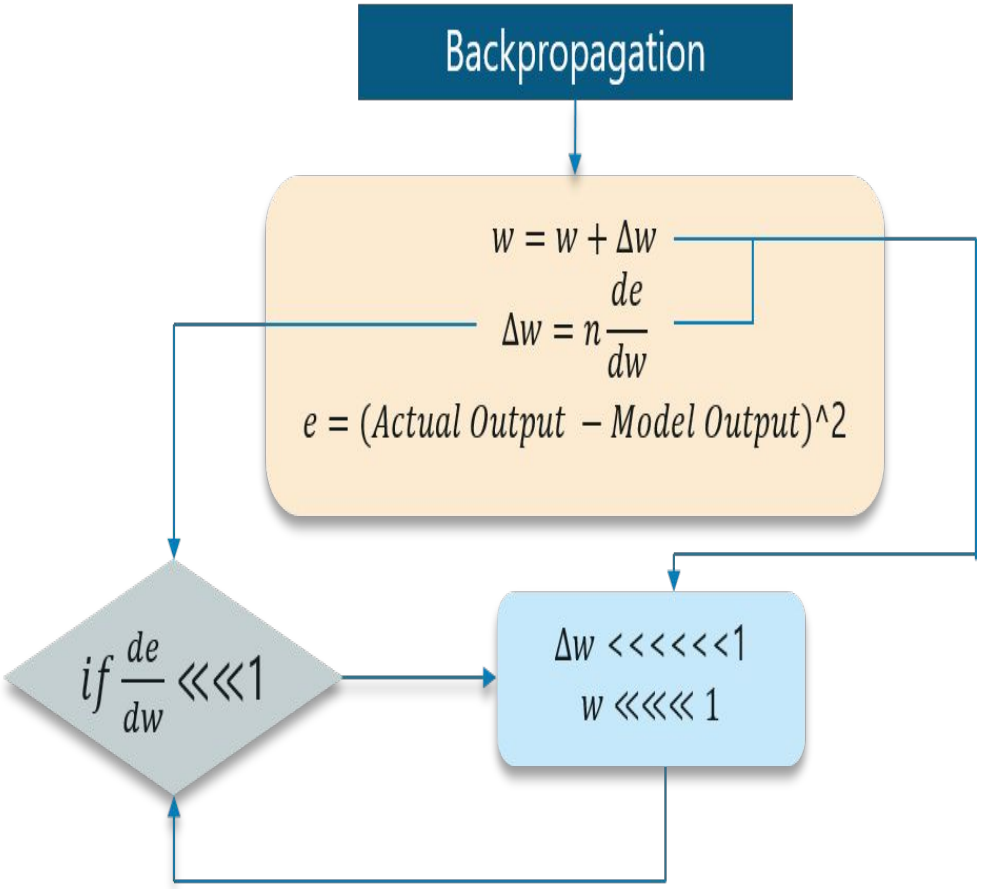
- To train the RNN, these weights need to updated.



- Recurrent Neural Networks use **backpropagation algorithm** for training, **but** it is **applied** for every **timestamp.** It is commonly known as **Back-propagation Through Time (BTT).**

Training RNN…

- There are **some issues** with Back-propagation such as:
  - ✔ **Vanishing Gradient**
  - ✔ **Exploding Gradient**
- **Vanishing Gradient**
  - ✓ In **back-propagation,** the **goal** is to **calculate** the **error** $(e) = (Actual\ Output - Model\ Output)^2$.
  - ✓ The **changes** in the e w.r.t the **change** in the **weight** is **calculated** and multiplied with learning rate (n), which is the **actual change** in the weight. This change in weight Δw is added to the **old set** of weight for every iterations.
  - ✓ The issue here is when change of e w.r.t. to w is multiplied with n, the value may be very less. It is also called as **vanishing gradient**. At this condition now further weights are updated.

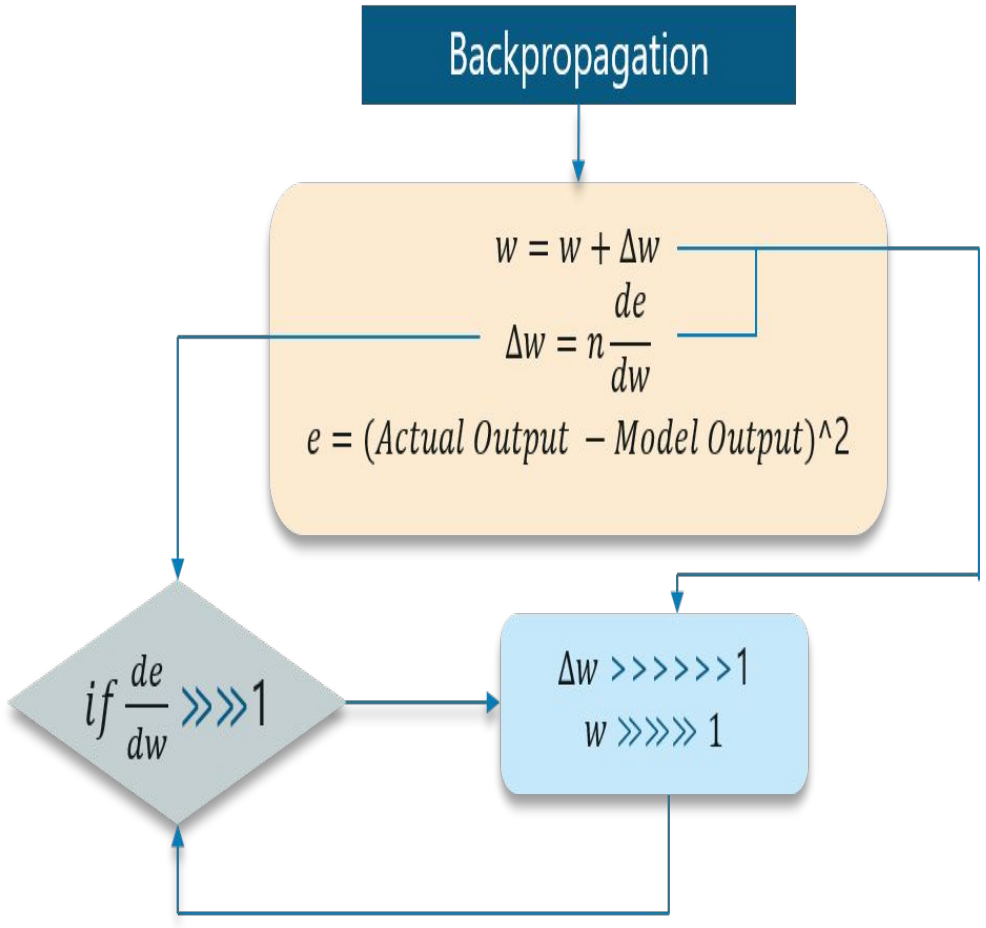


Training RNN…

- **Exploding Gradient**
  - ✔ The working of the exploding gradient is **similar** but the **weights here** change **drastically** instead of **negligible change.**
  - ✔ To overcome with these issues, following may be done.

Issues and Solutions

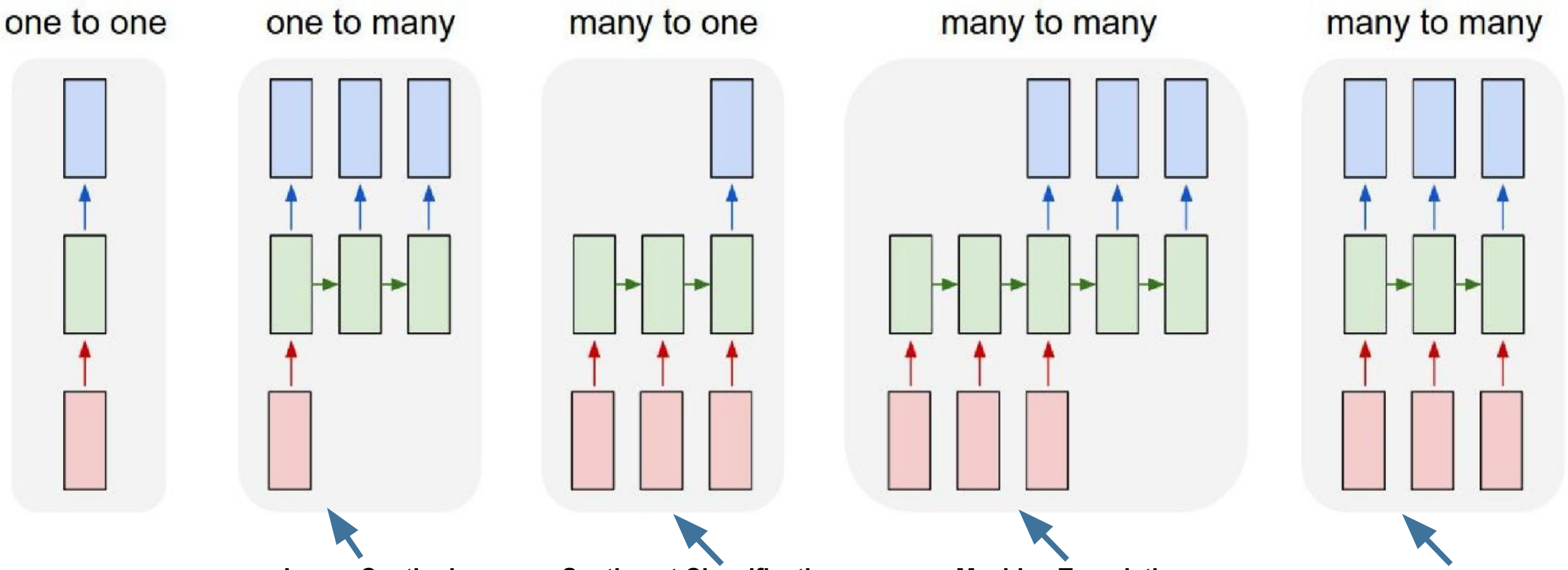
- **Exploding Gradient**
  - ✔ Clip the gradient using threshold.
  - ✔ Using RMSprop optimizer, to adjust the learning rate.
- **Vanishing Gradient**
  - ✔ Use ReLU activation function.
  - ✔ Different network can be used such as LSTM and GRU.



Applications

- **Recognize patterns** in sequences of data such as:
  - ✔ Stock Market prediction
  - ✔ Text Generation
  - ✔ Genomes
  - ✔ Handwriting,
  - ✔ Spoken word
  - ✔ Numerical times **series** data emanating from sensors

Applications



References

- https://www.youtube.com/watch?v=UNmqTiOnRfg
- https://www.youtube.com/watch?v=WCUNPb-5EYI
- https://www.youtube.com/watch?v=iX5V1WpxxkY
- https://www.edureka.co/blog/recurrent-neural-networks/

Thank You

Contact: dinesh@dtu.ac.in
Mobile: +91-9971339840