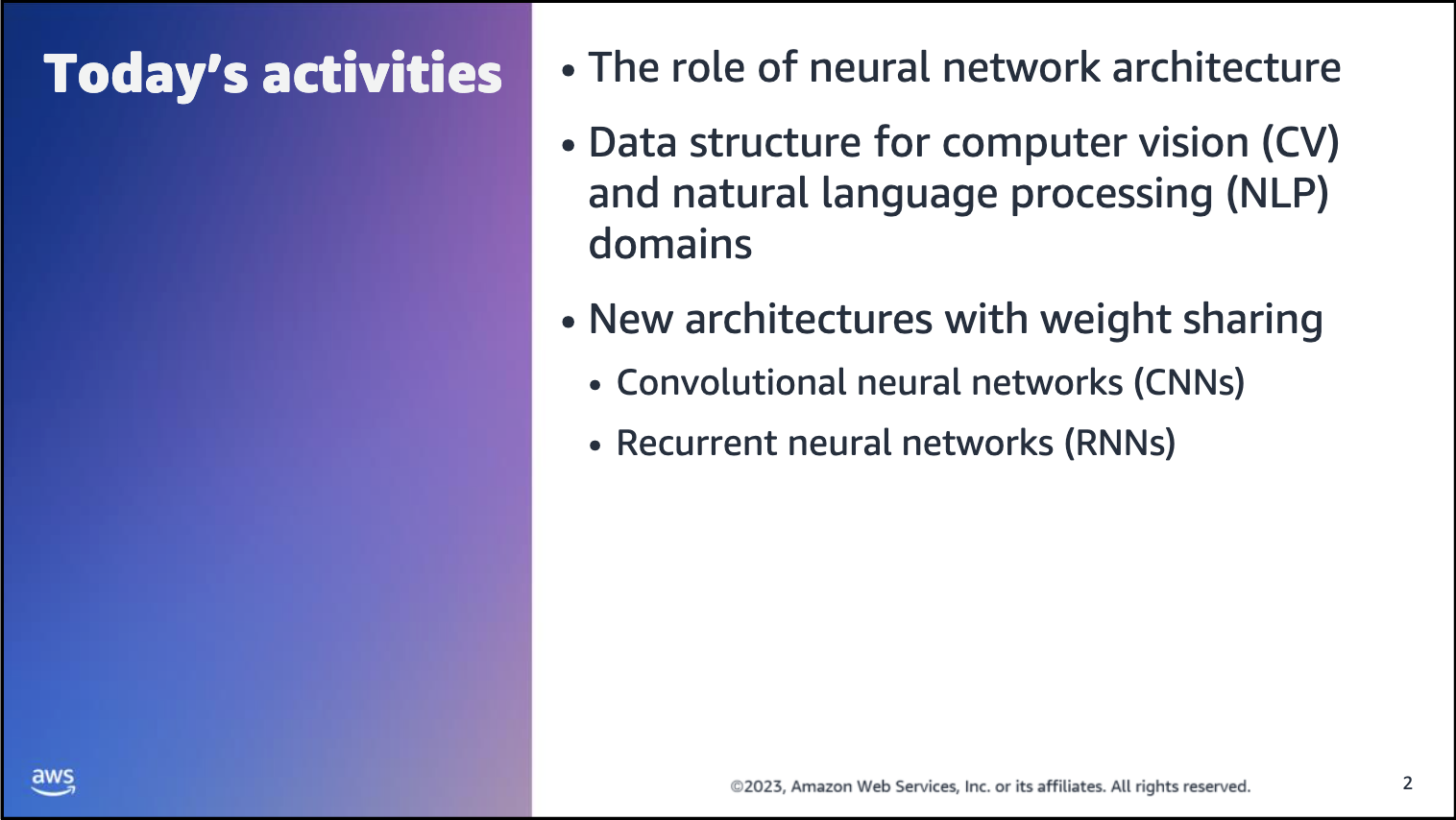
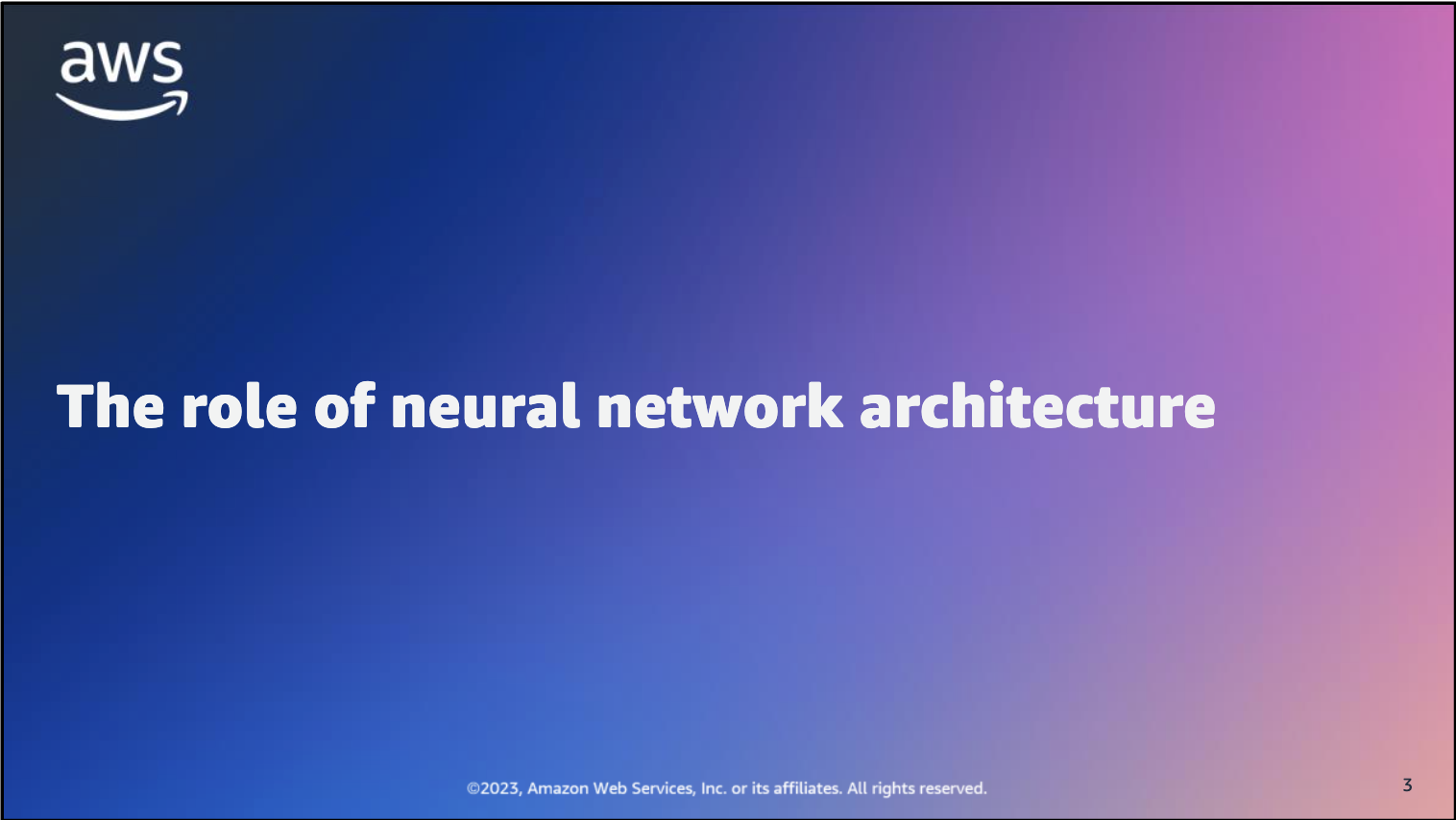
****

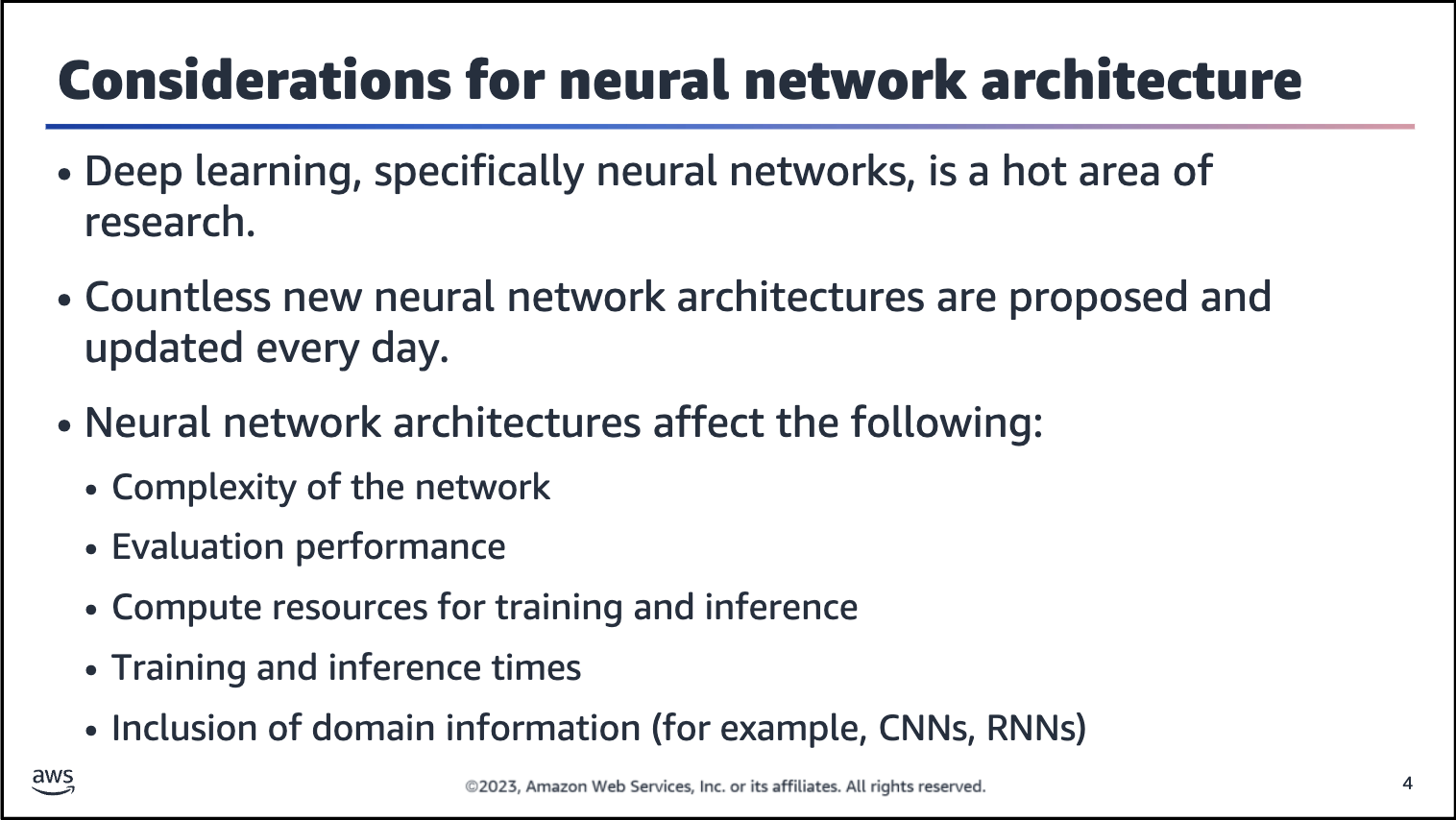
**Application of Deep Learning to Text and Image Data**

Module 1 Lesson 4 Student Guide









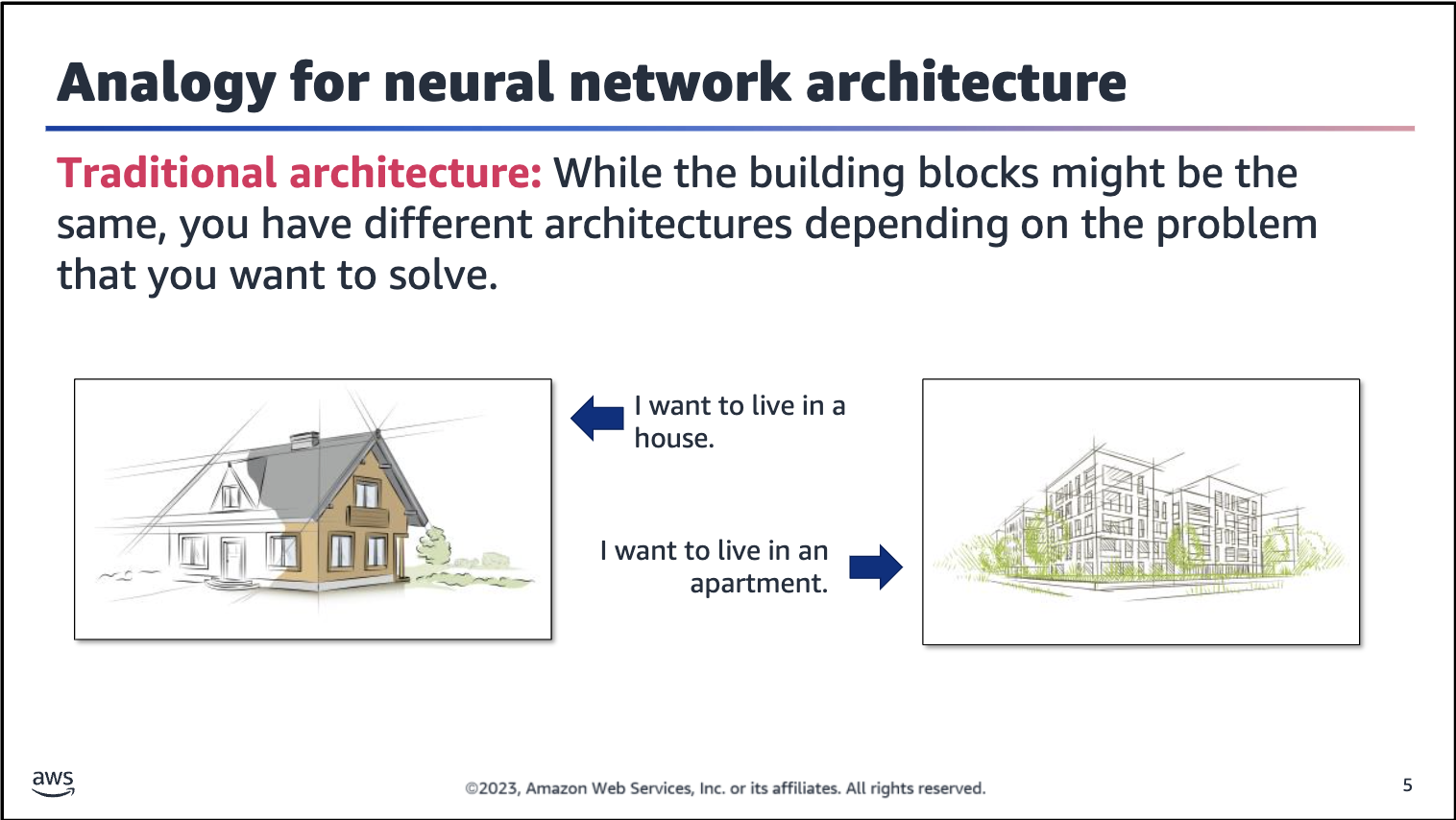
Over the past few years, the field of deep learning has attracted a lot of interest. Neural networks make up an integral part of the deep learning process. The primary goal of neural networks is to transform input into a richer, more meaningful representation.

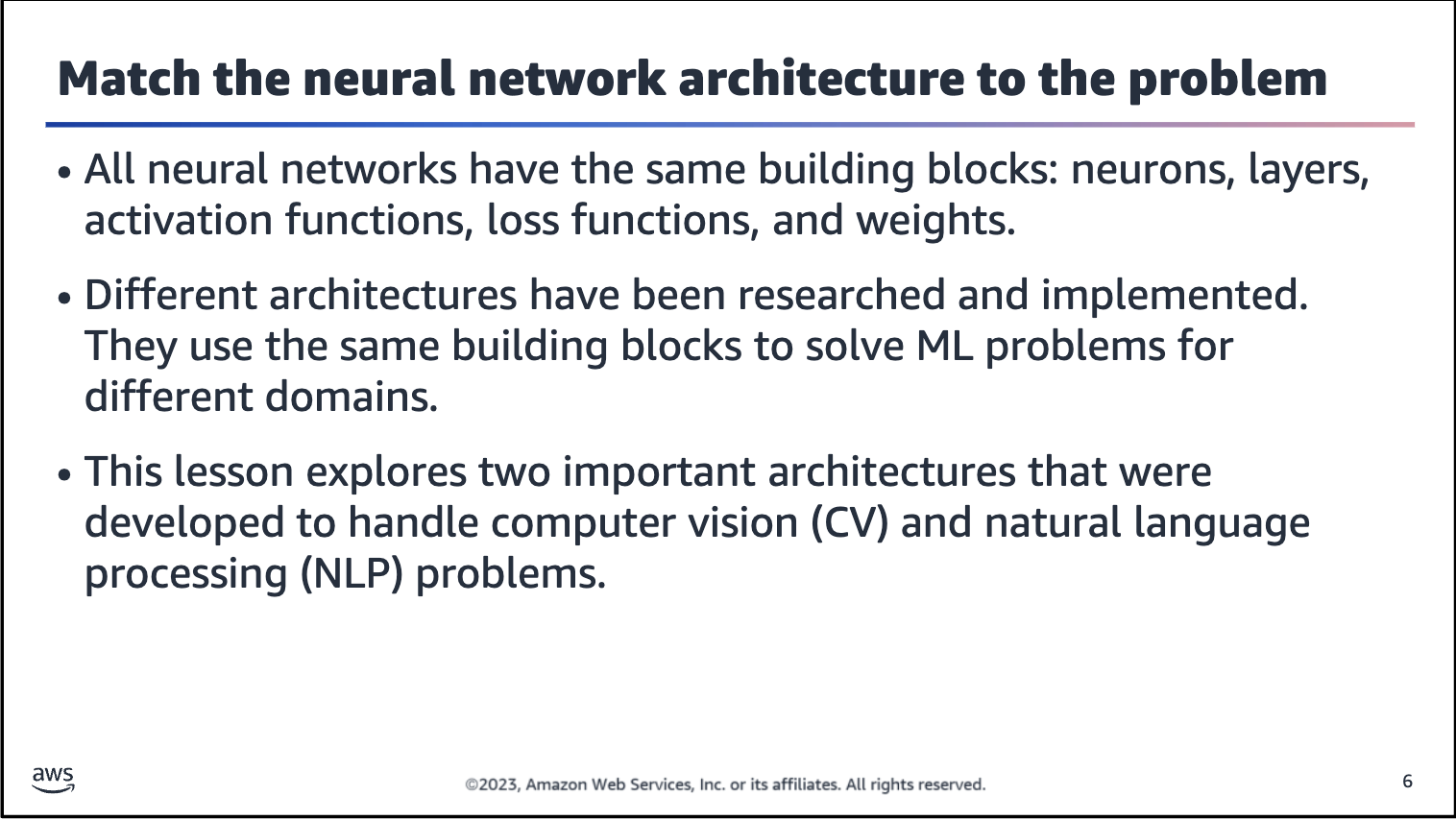
As discussed previously, a neural network consists of an input layer, an output layer, and one or more hidden layer. The design and architecture of a feed-forward neural network primarily depends on the number of neurons in each layer and the number

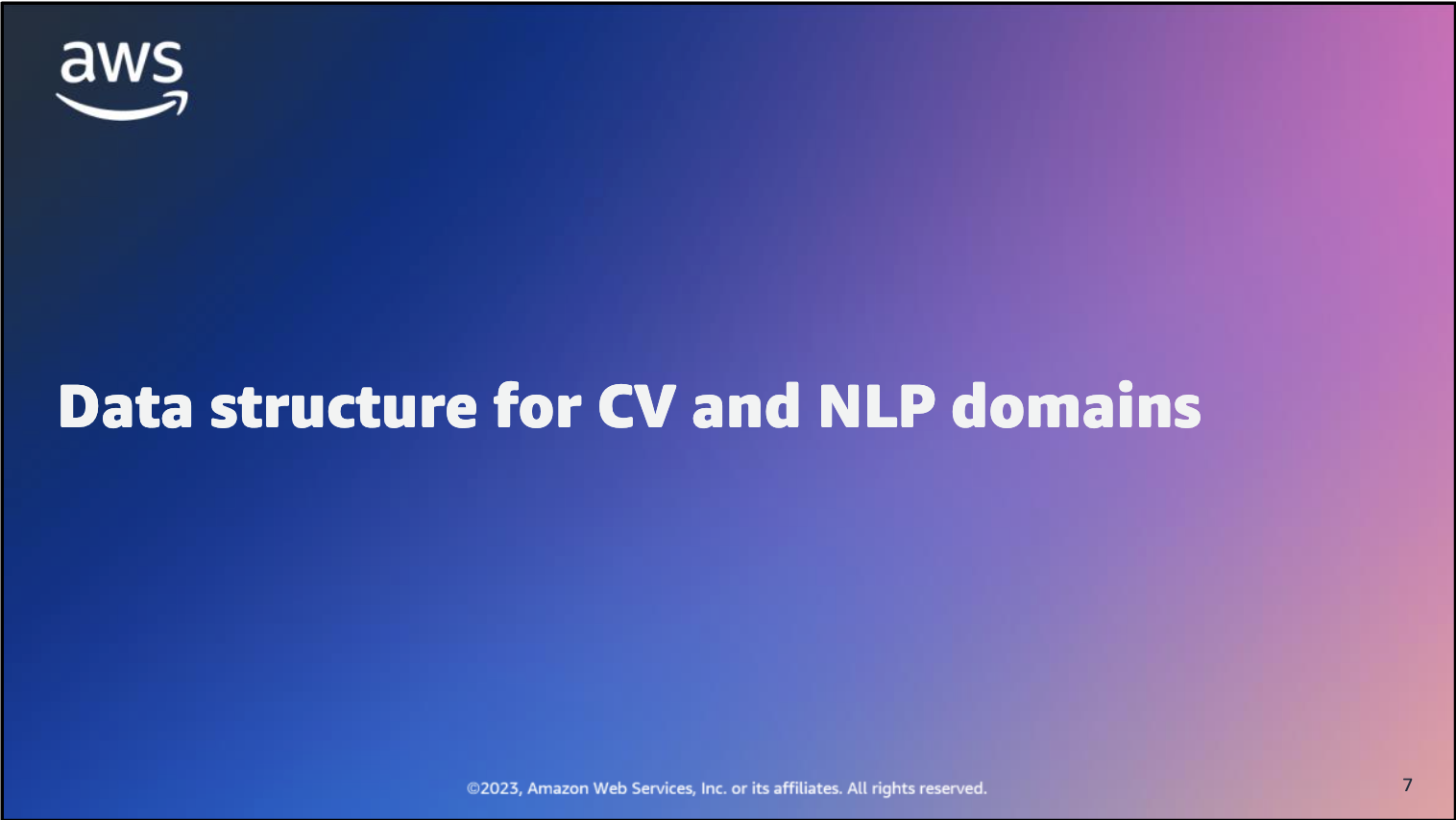
of hidden layers.

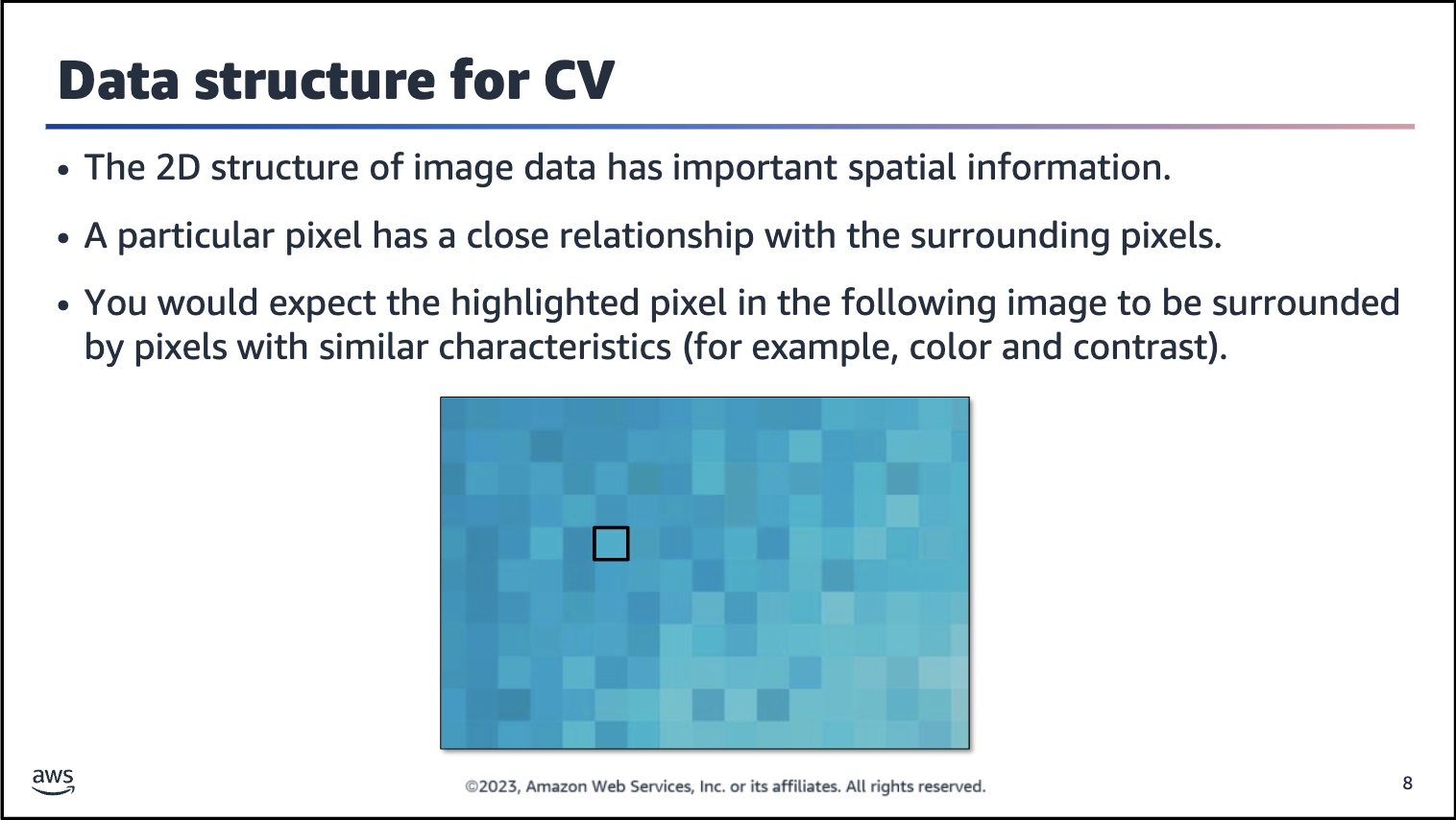
As the field evolved, neural networks were designed to meet the needs of specialized domains, such as convolutional neural networks (CNNs) for image data and recurrent neural networks (RNNs) for sequential data.

The architecture of a neural network often plays a crucial role in the overall performance of the network. The architecture also affects the resources — both compute and time — that are required to train and predict the network.

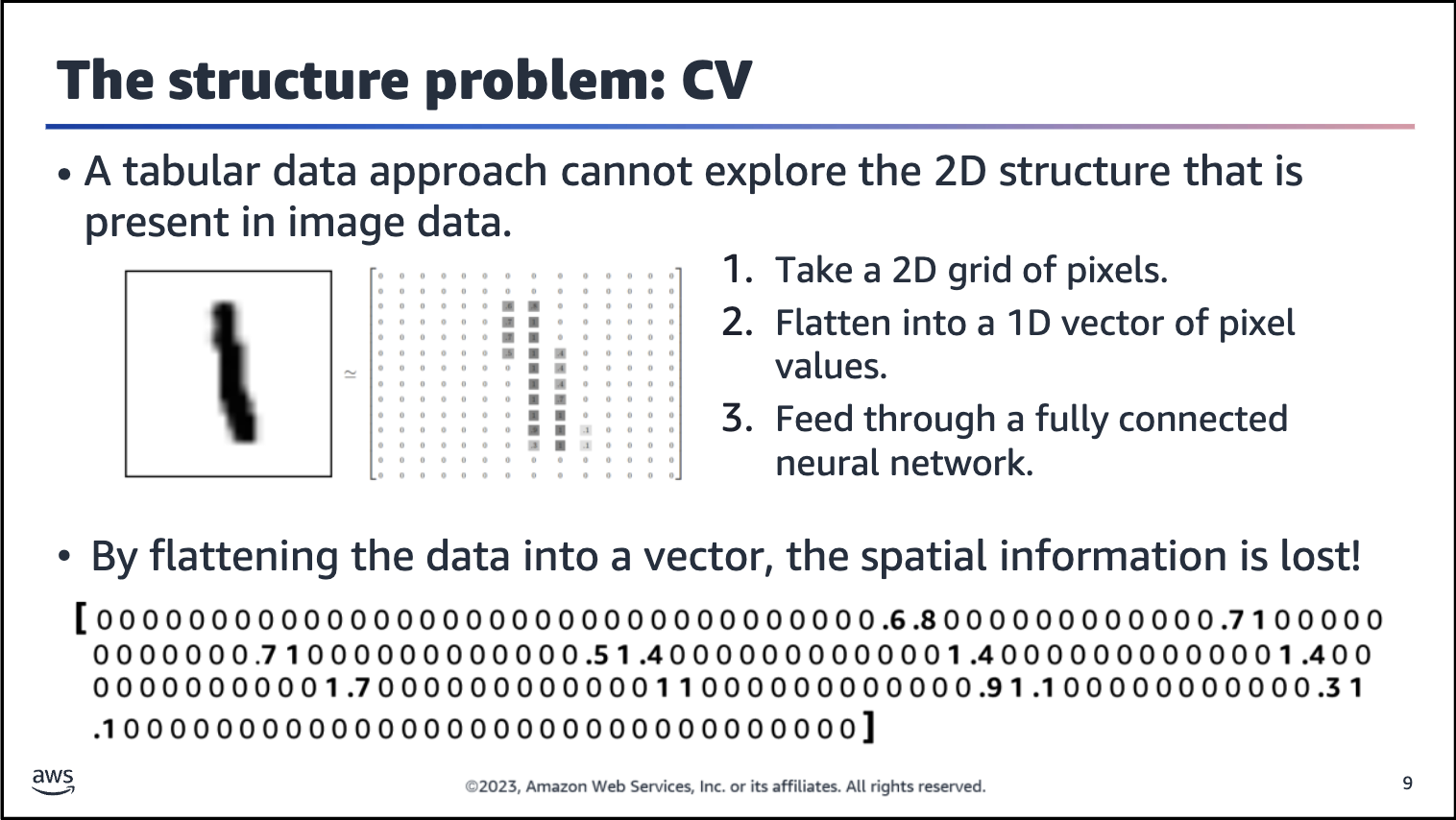








The assumption that the surrounding pixels would have similar characteristics holds true for most images because a significantly different pixel color, such as bright yellow, would seem like an error. A similar concept is used to extract information from images.



This slide demonstrates how a tabular data approach cannot capture the 2D structure of an image.

With image data, the pixel values become the features. Each cell in the matrix represents a grayscale number that is related to each pixel (0 represents white, 1 represents black, and intermediate numbers represent shades of gray).

To use these values as input for a model, you need to flatten this matrix into a single-dimension vector. This is because fully connected neural networks models, which are also called multilayer perceptrons (MLPs), can only receive vectors as input. In the vector image, the pixels are now distributed sparsely and have lost most of the information that is related to their 2D position. Even worse, fully connected neural network models don’t capture the spatial arrangement of pixels in an image.

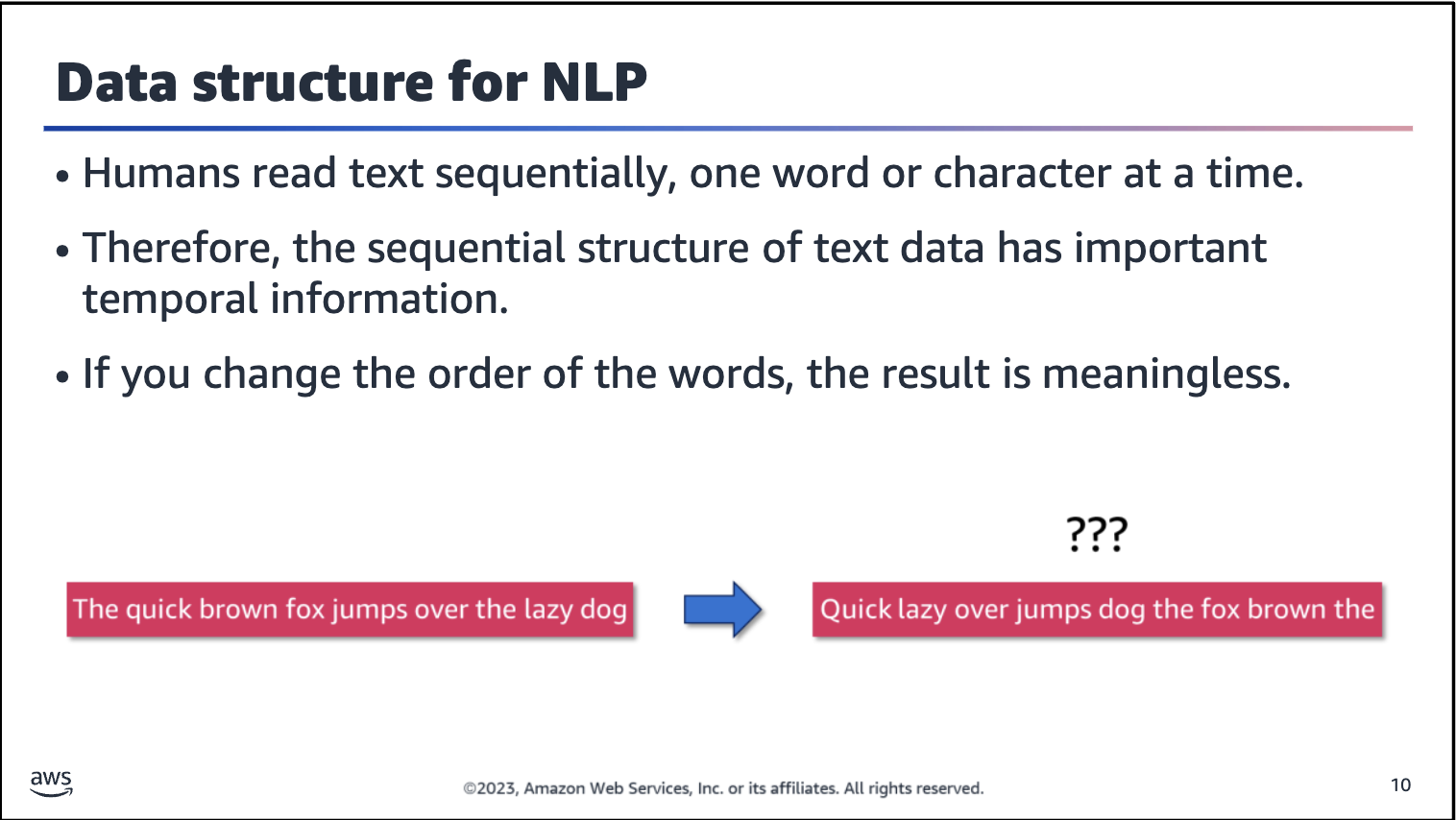


Image description: The sentence "The quick brown fox jumps over the lazy dog" is transformed into a random order of "Quick lazy over jumps dog the fox brown the." This example highlights how the sequence of text matters.

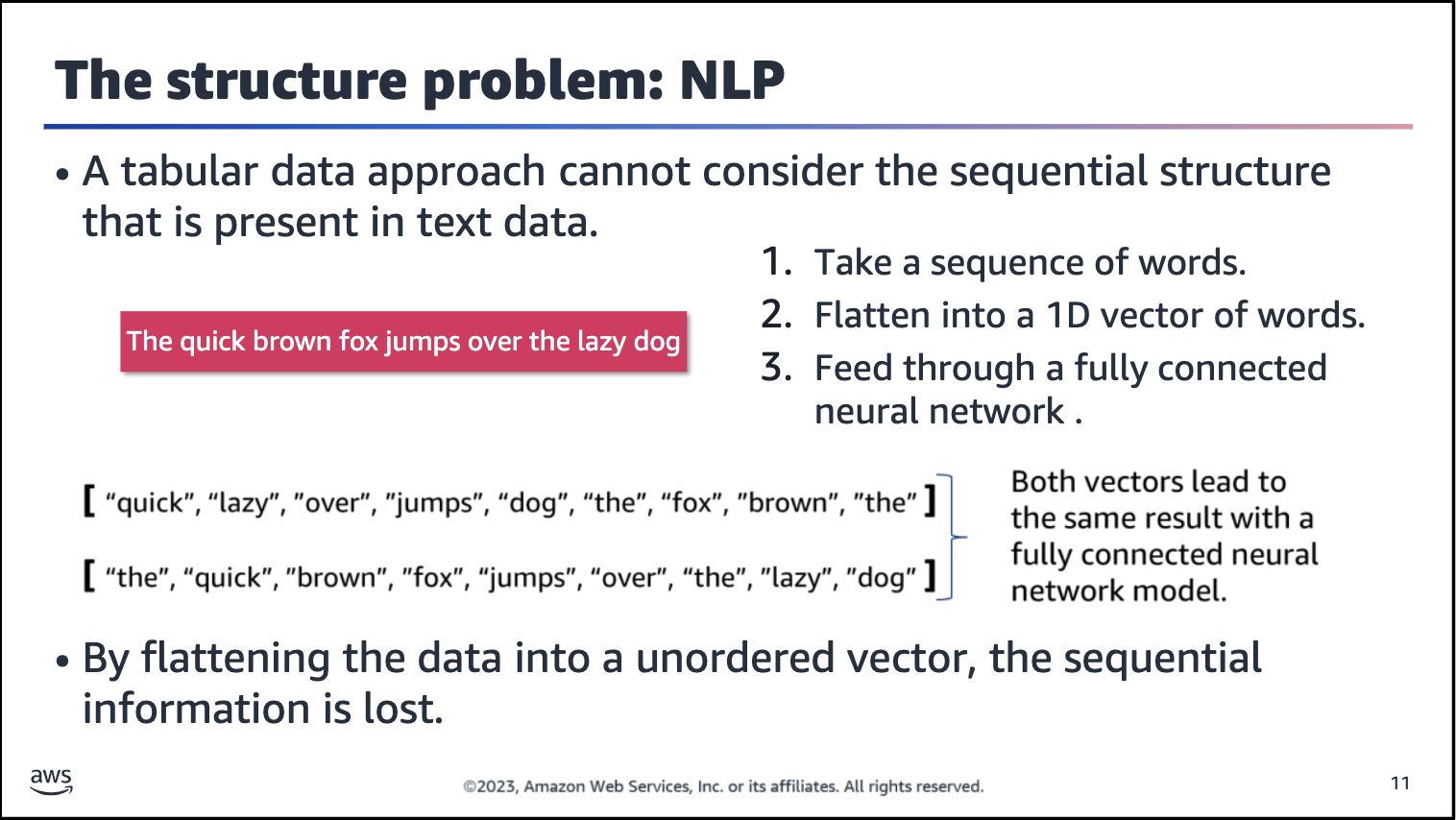
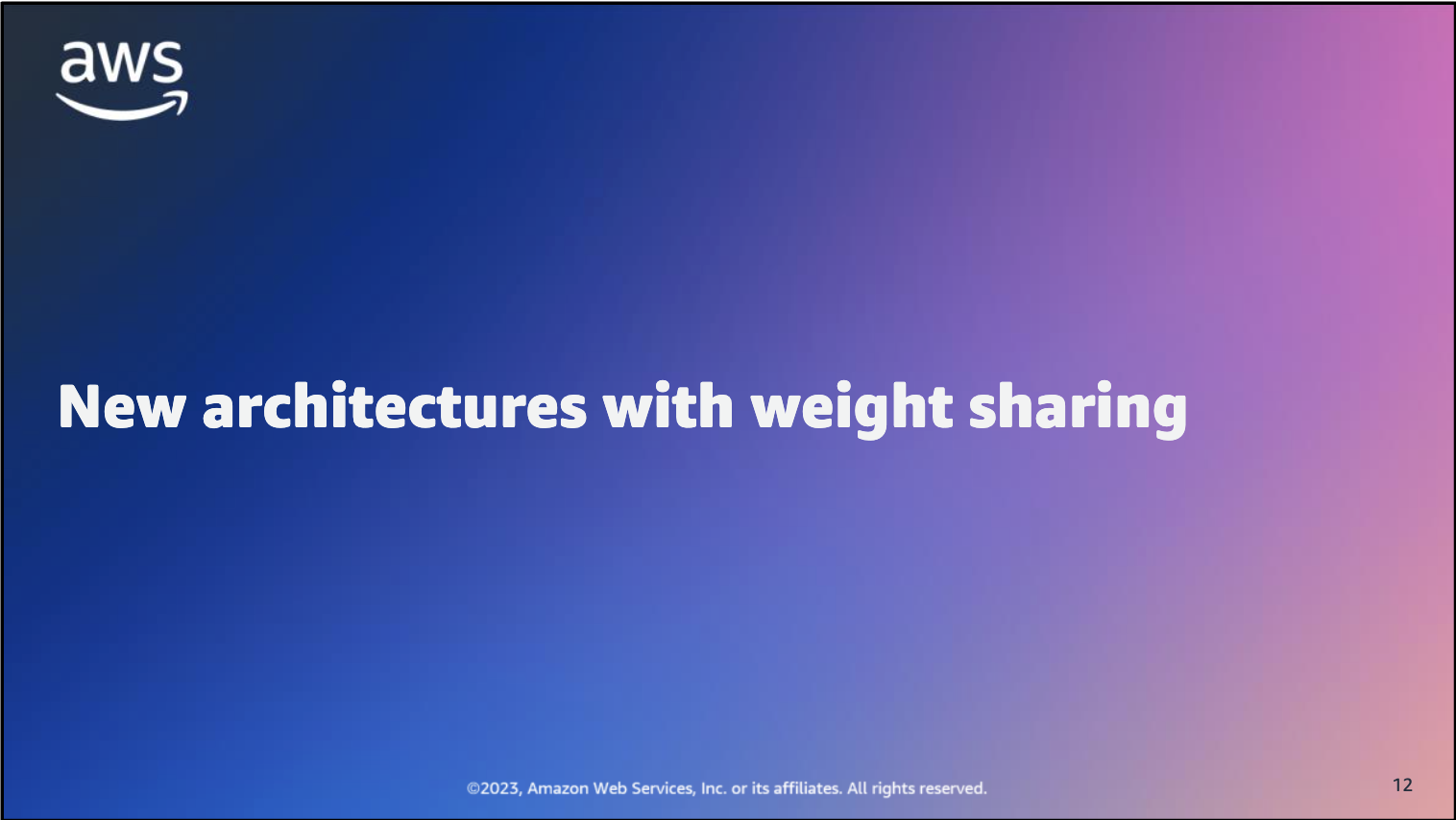
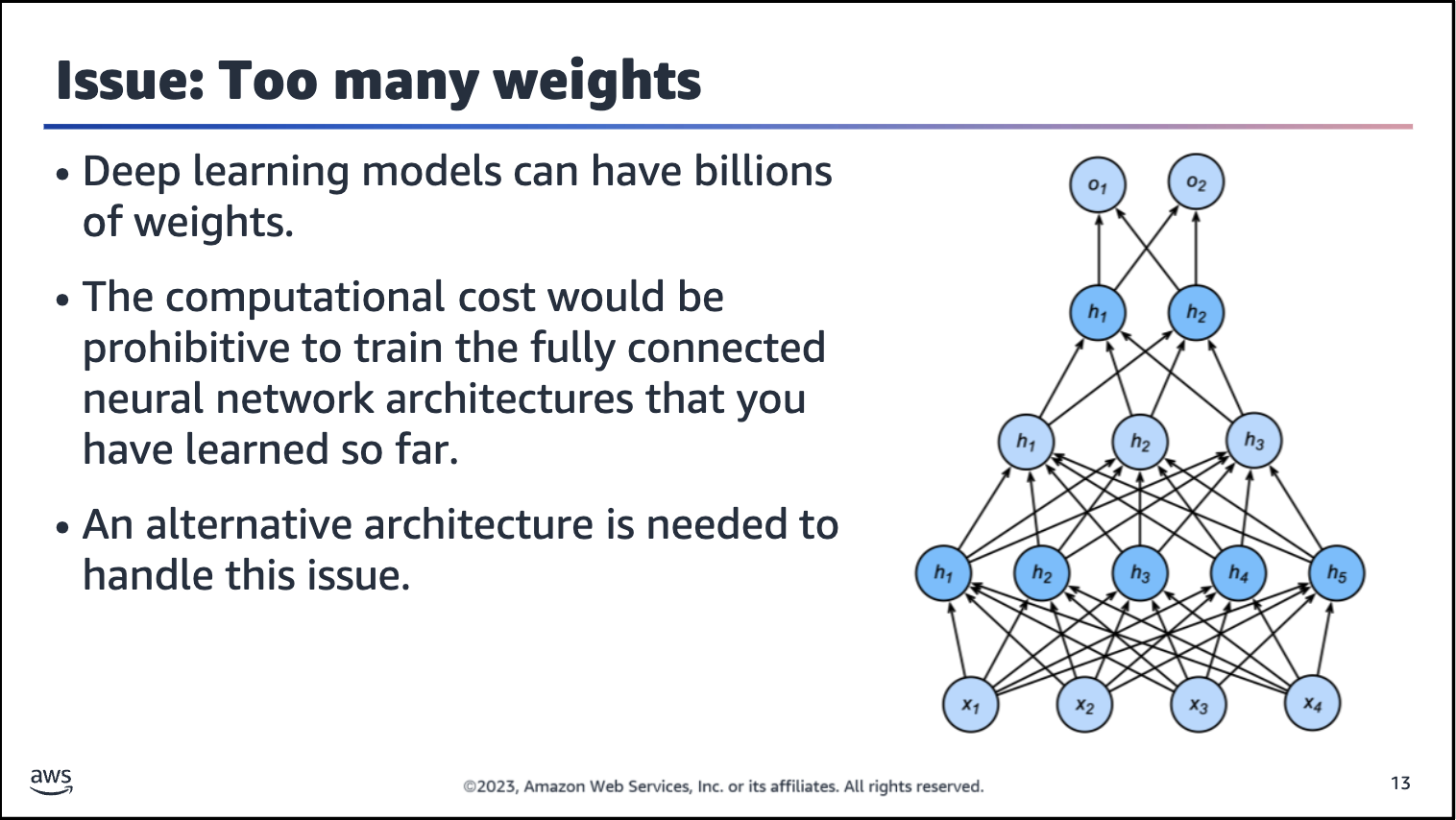
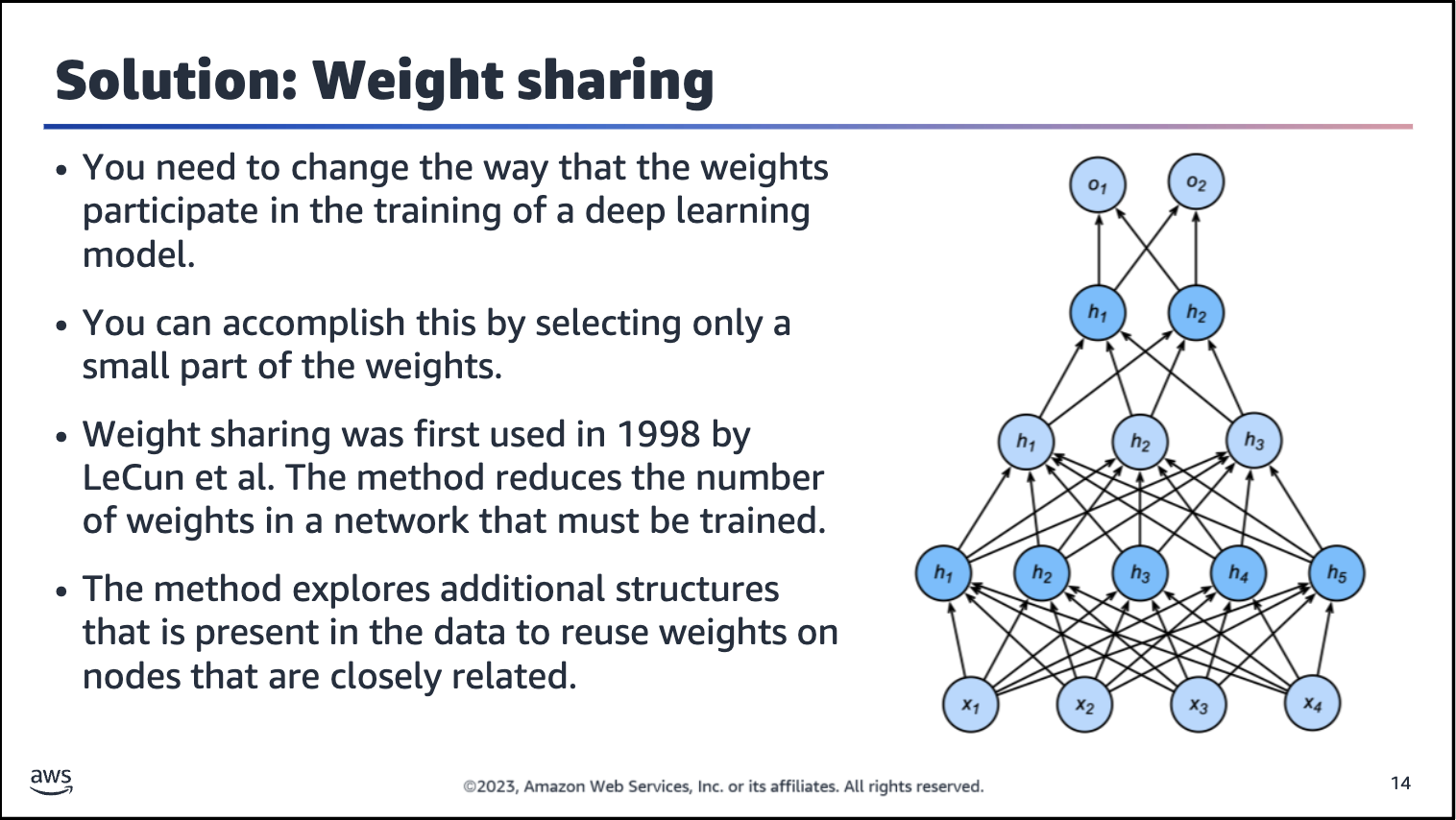


Image description: Two vectors contain the words from the sentence “The quick brown fox jumps over the lazy dog,” but in different order. Both vectors would lead to the same result with a fully connected neural network model because order is not

considered.

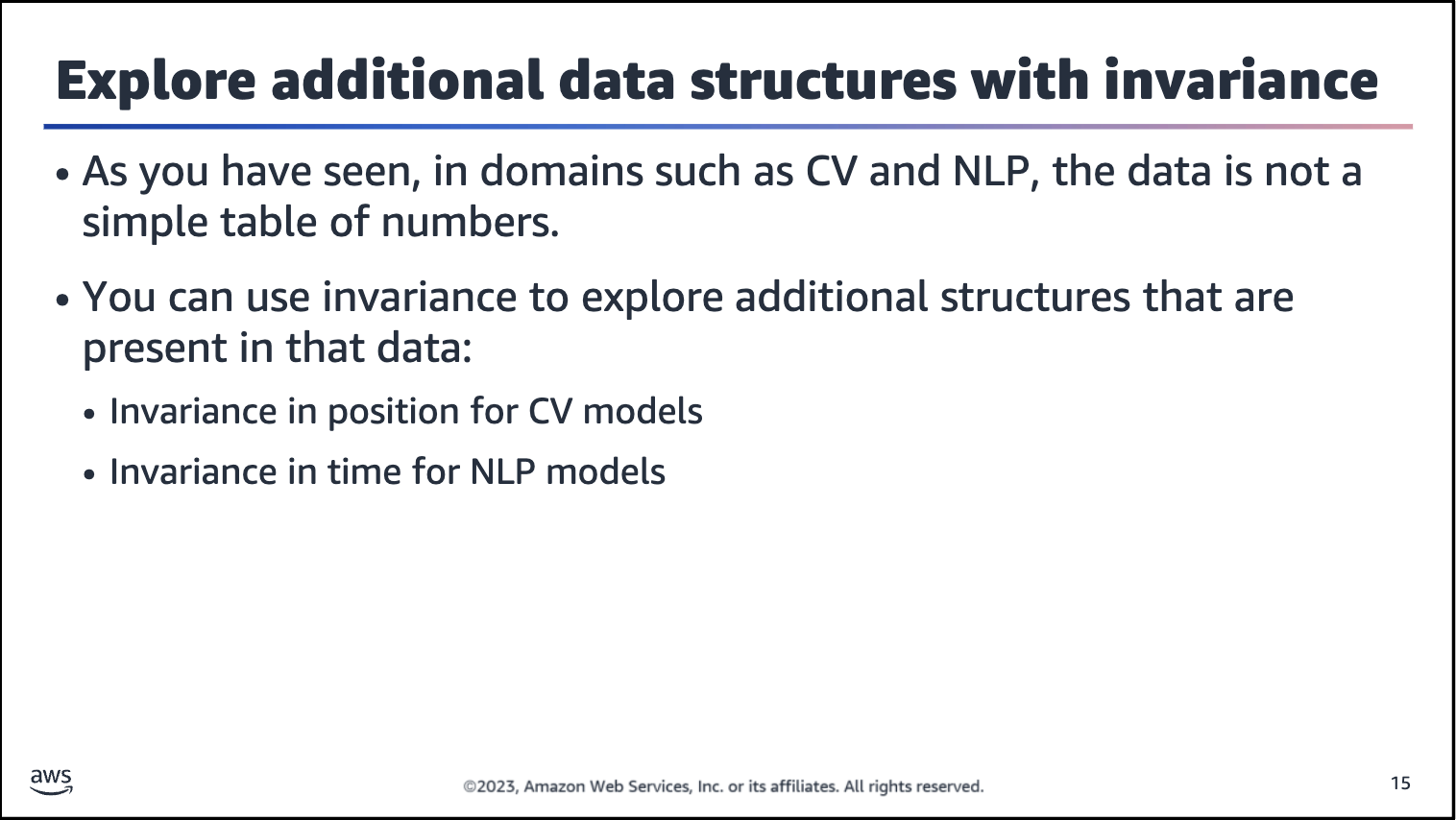






Reference: “LeNet.” Wikipedia, June 16, 2023.

<https://en.wikipedia.org/wiki/LeNet>.



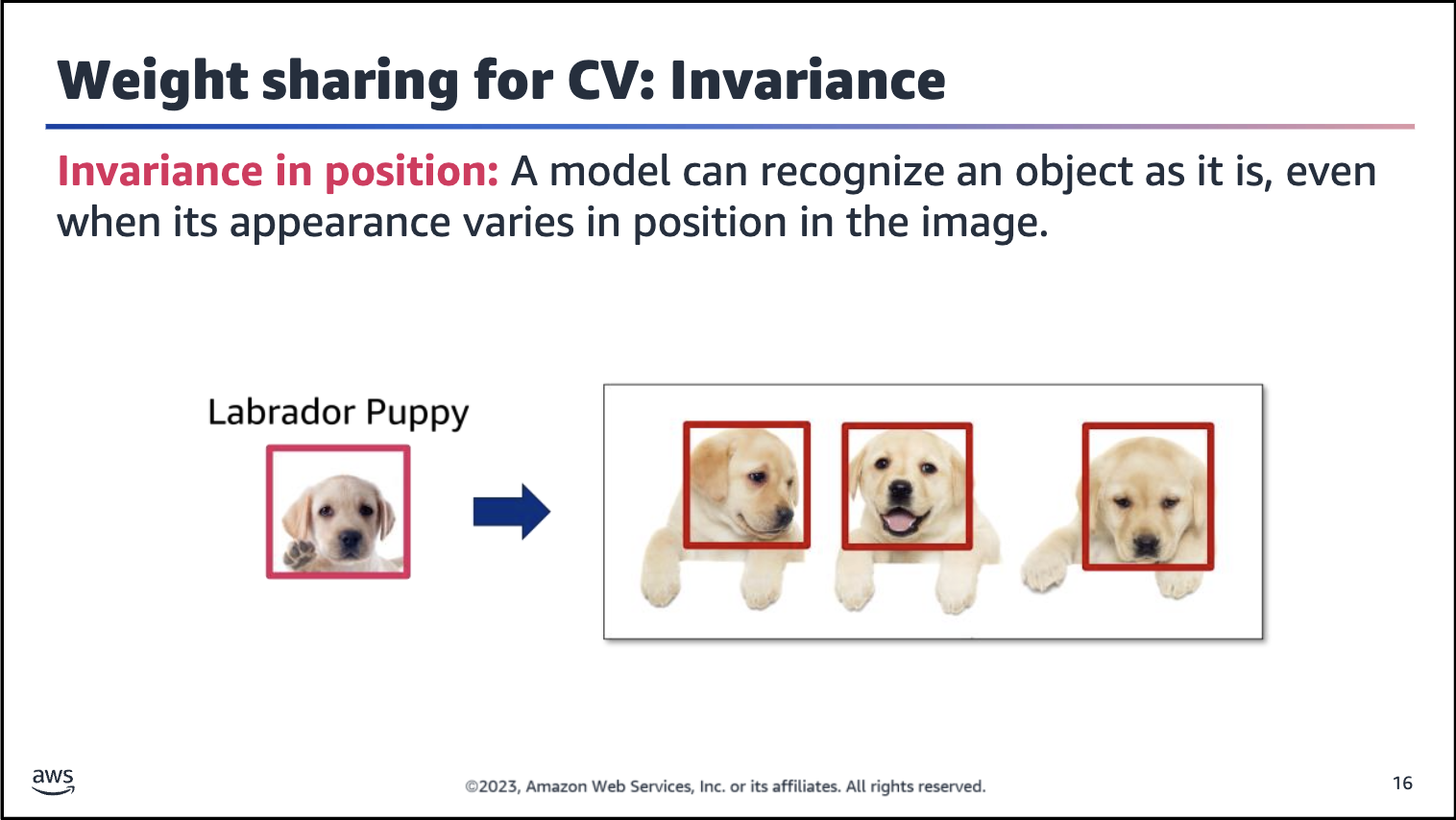


Image description: The first image shows a Labrador puppy with a rectangle drawn around its head. The second image shows three Labrador puppies in different positions, each with a rectangle drawn around its head.

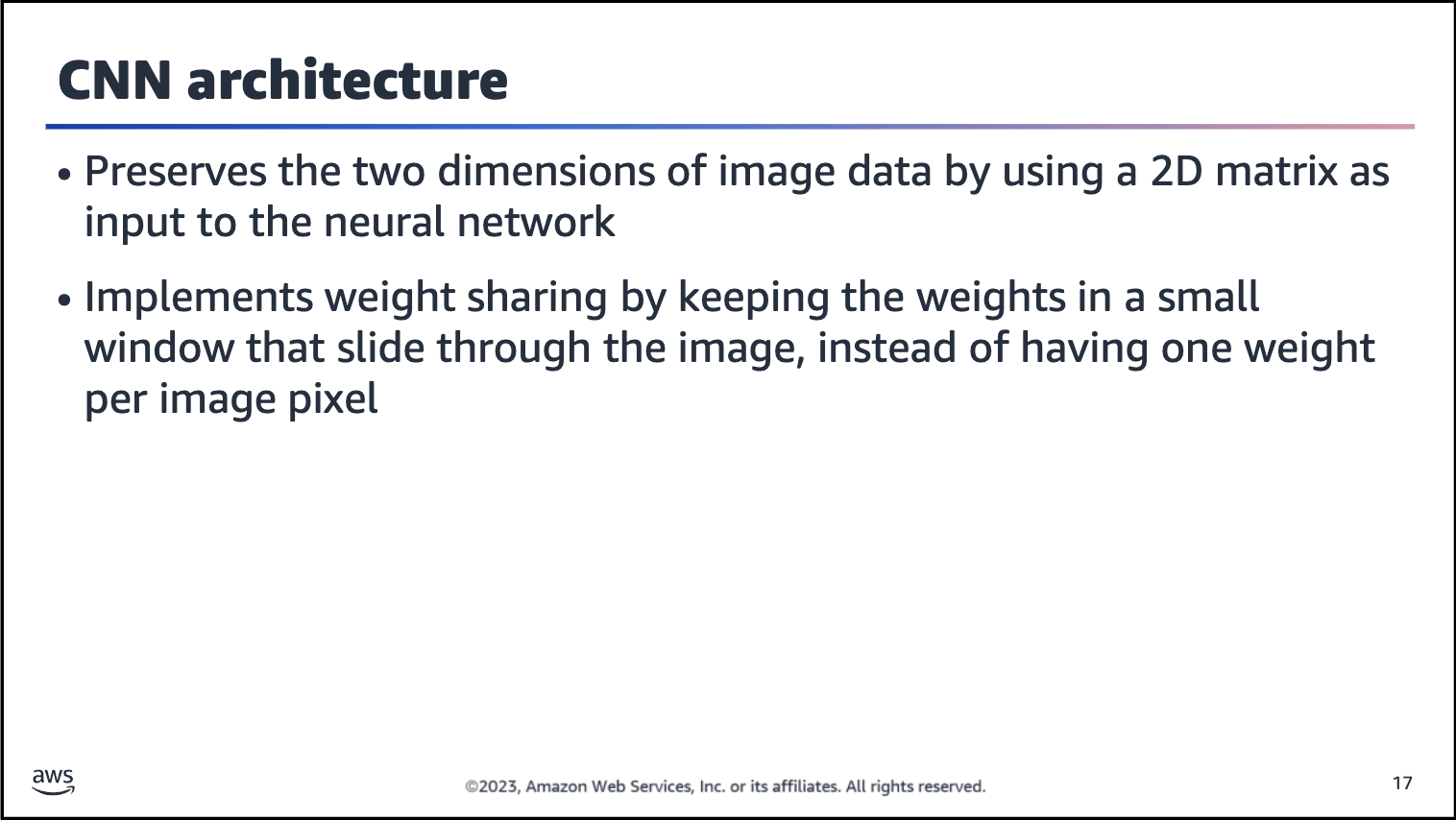
*Invariance in position* is a way that a model can recognize an object as an object, even when its appearance varies in position in the image. The example highlights that, no matter where the Labrador puppy appears in the image, the model can detect the

main characteristics of the puppy and detect it in the image.

Analogy with your brain: After you looked at the puppy in the left image and then moved to examine the image on the right, the three puppies popped out immediately. That is because your brain explores this invariance in position, transporting the same general characteristics of the Labrador puppy that you saw in

the first image to the second image, so you were able to detect this pattern easily.

In other words, a model, like your brain, can look for these patterns and doesn’t need to fully inspect the image before detecting the patterns that are related to the first image.



In the module about CV, you will learn how the CNN architecture works.

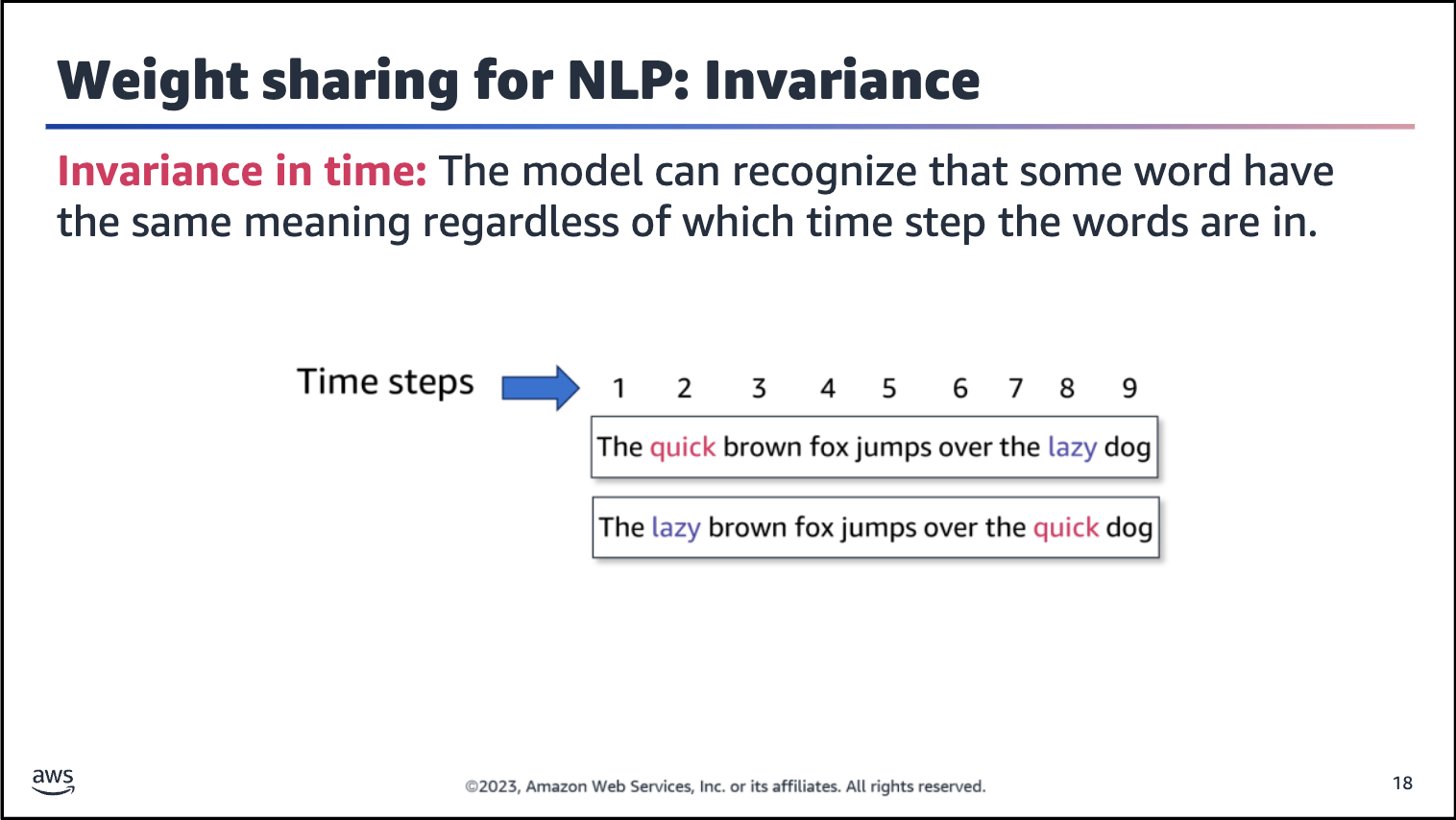


Image description: Two versions of the same sentence, where the words “quick” and “lazy” are swapped between them. Sentence 1: The quick brown fox jumps over the lazy dog. Sentence 2: The lazy brown fox jumps over the quick dog. Above the sentences is a row of numbers from 1 to 9, with one number above each word. The numbers indicate the time step that each word is in.

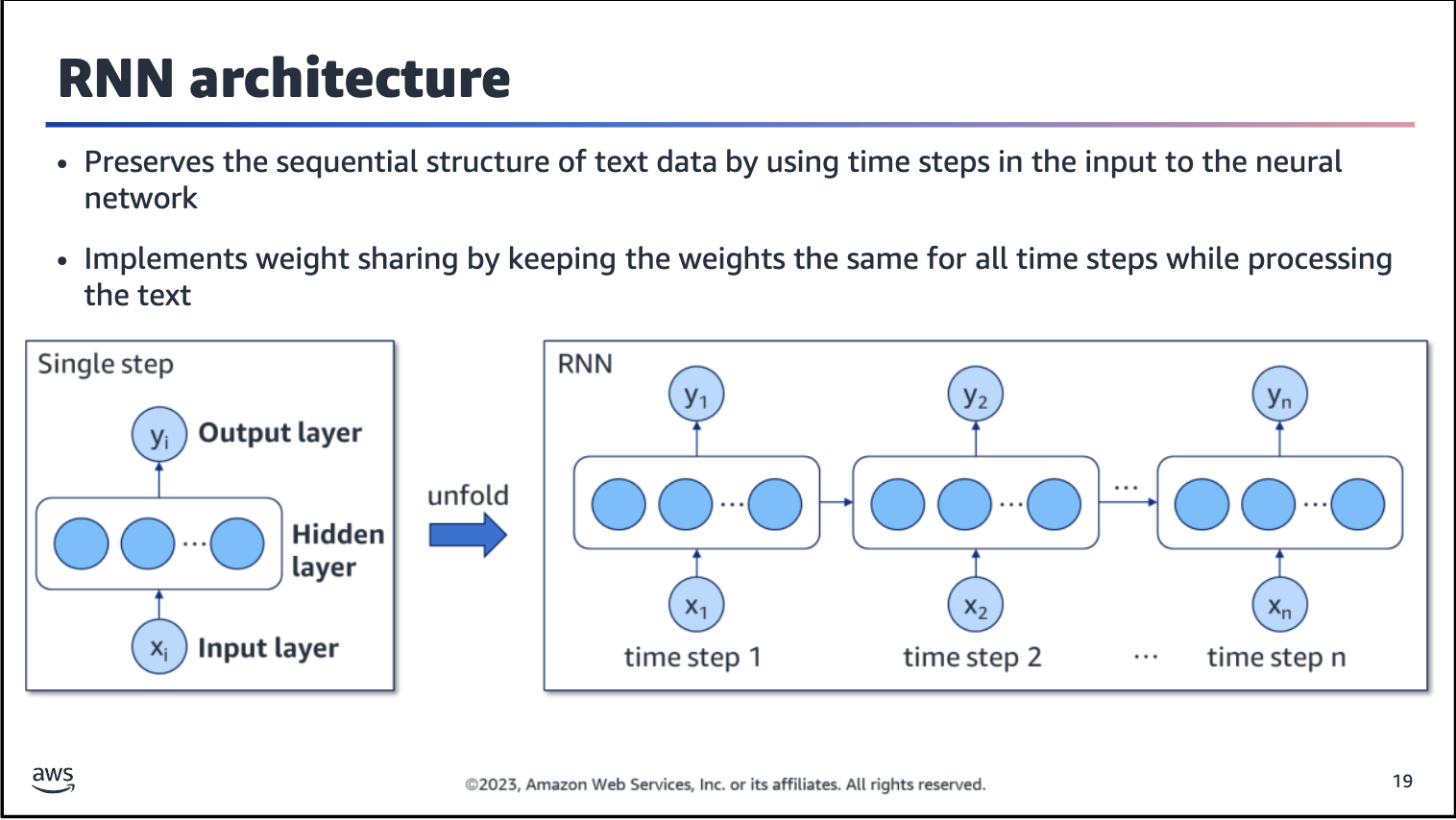


Image description: One side of the illustration shows the key components of a single step in an RNN. The components are input layer, hidden layer, and output layer. The word “unfold” connects to the other side of the illustration. This side shows how a single time step unfolds into many time steps when data passes through an RNN architecture.

You will learn how it works in the module about NLP.

