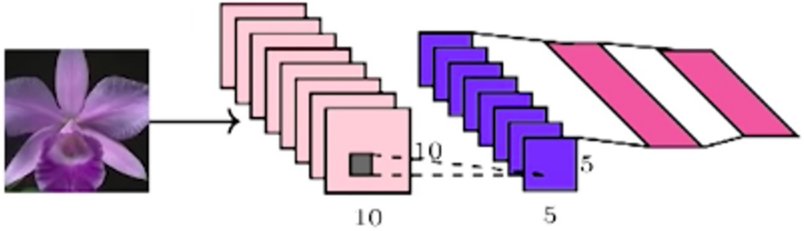
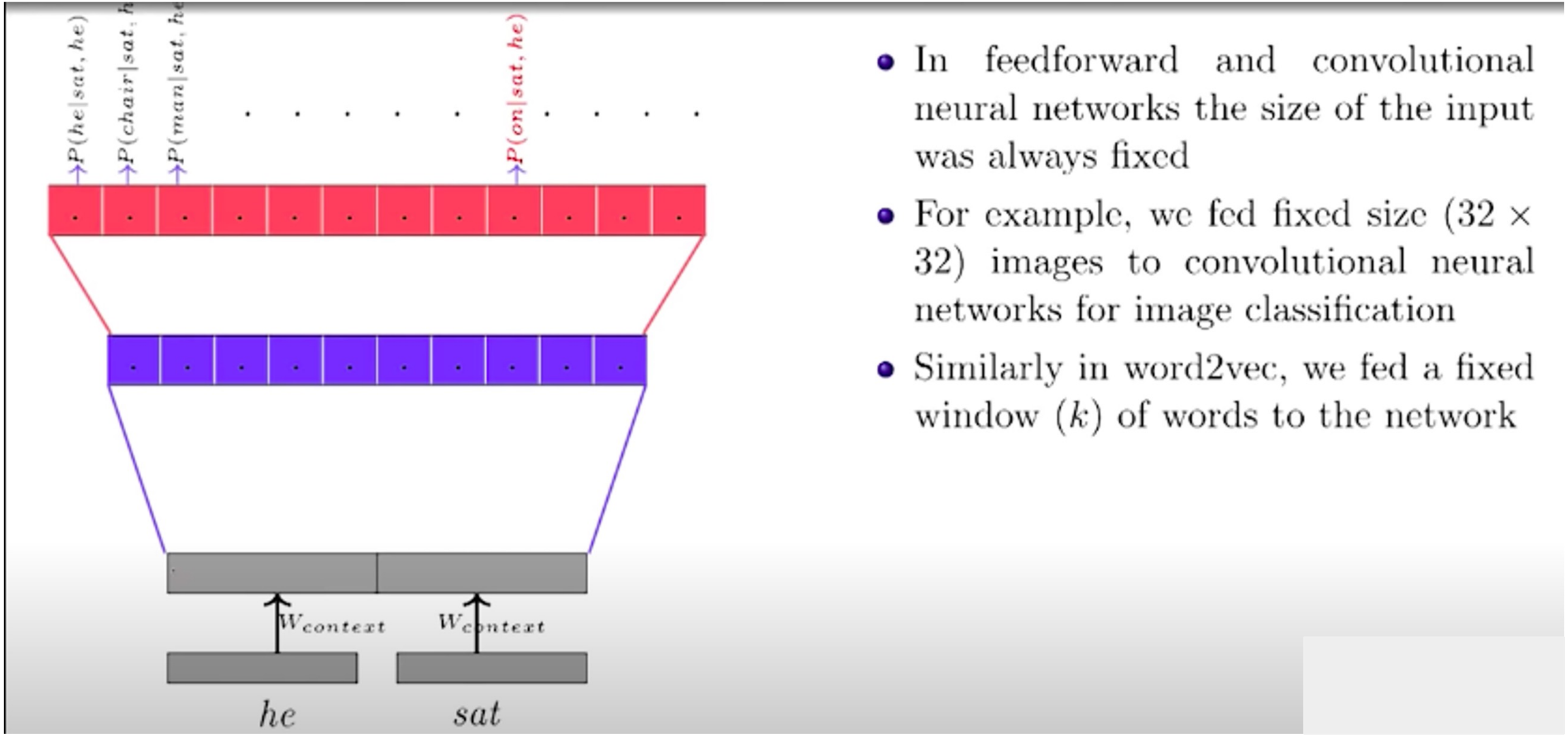
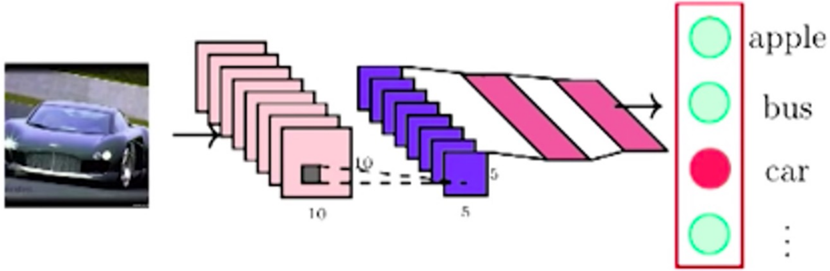
Sequence Learning Problems



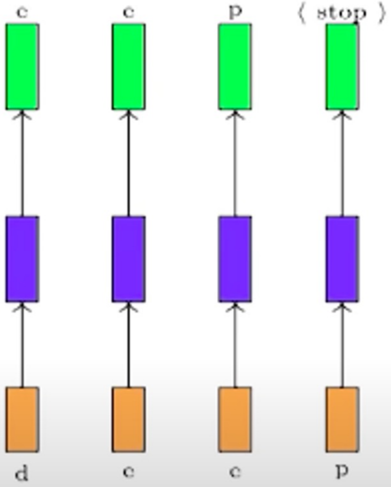
- In feedforward and convolutional neural networks the size of the input was always fixed
- For example, we fed fixed size (32 × 32) images to convolutional neural networks for image classification



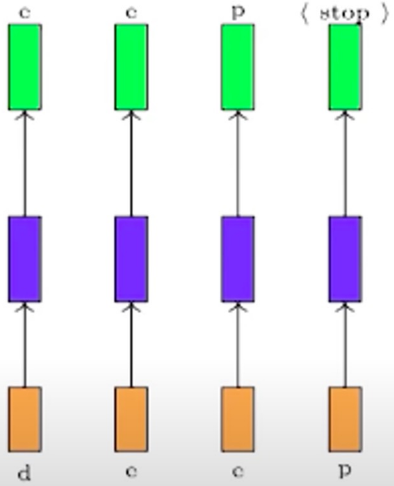
- In feedforward and convolutional neural networks the size of the input was always fixed
- For example, we fed fixed size (32 × 32) images to convolutional neural networks for image classification
- Similarly in word2vec, we fed a fixed window (k) of words to the network



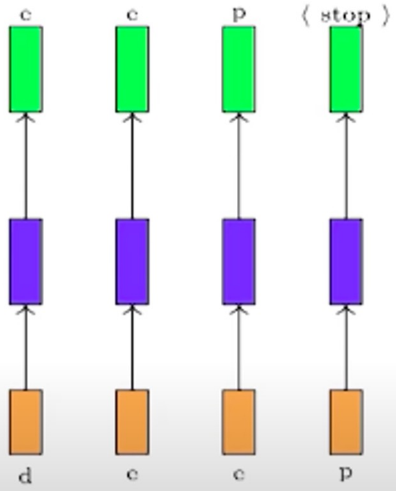
- In feedforward and convolutional neural networks the size of the input was always fixed
- For example, we fed fixed size (32 × 32) images to convolutional neural networks for image classification
- Similarly in word2vec, we fed a fixed window (k) of words to the network
- Further, each input to the network was independent of the previous or future inputs
- For example, the computatations, outputs and decisions for two successive images are completely independent of each other



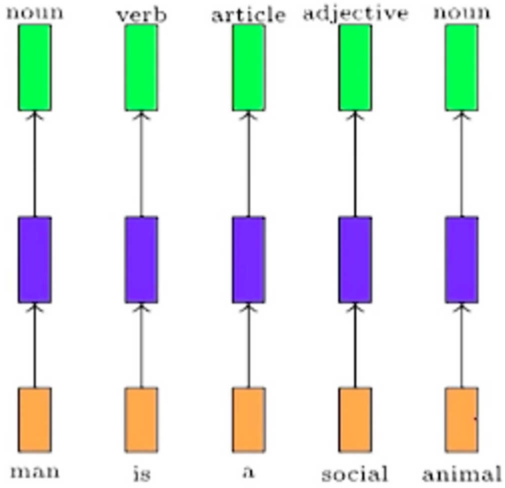
- In many applications the input is not of a fixed size
- Further successive inputs may not be independent of each other
- For example, consider the task of auto completion
- Given the first character ‘d’ you want to predict the next character ‘e’ and so on



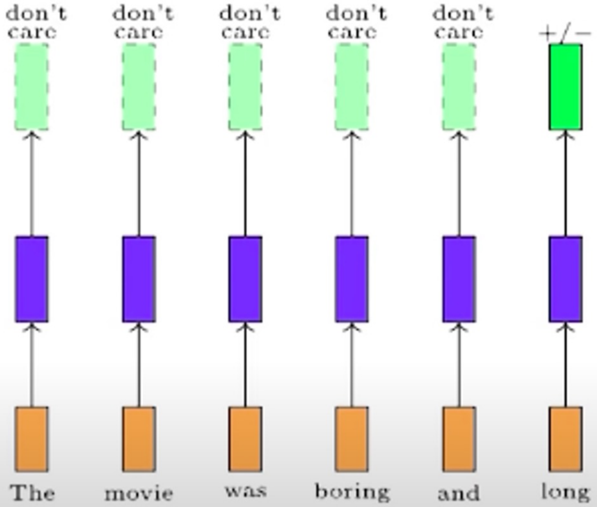
- Notice a few things
- First, successive inputs are no longer independent (while predicting 'e' you would want to know what the previous input was in addition to the current input)
- Second, the length of the inputs and the number of predictions you need to make is not fixed (for example, "learn", "deep", "machine" have different number of characters)
- Third, each network (orange-blue-green structure) is performing the same task (**input** : character **output** : character)



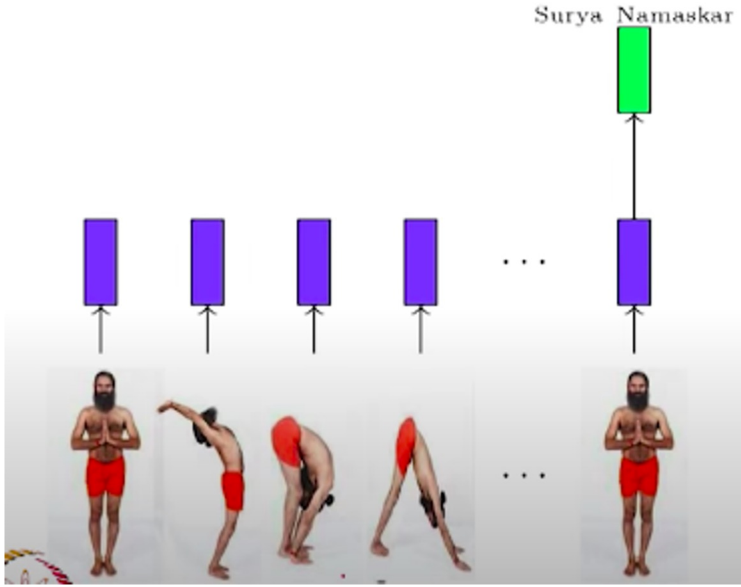
- These are known as sequence learning problems
- We need to look at a sequence of (dependent) inputs and produce an output (or outputs)
- Each input corresponds to one time step
- Let us look at some more examples of such problems



- Consider the task of predicting the part of speech tag (noun, adverb, adjective verb) of each word in a sentence
- Once we see an adjective (social) we are almost sure that the next word should be a noun (man)
- Thus the current output (noun) depends on the current input as well as the previous input
- Further the size of the input is not fixed (sentences could have arbitrary number of words)
- Notice that here we are interested in producing an output at each time step
- Each network is performing the same task (**input** : word, **output** : tag)



- Sometimes we may not be interested in producing an output at every stage
- Instead we would look at the full sequence and then produce an output
- For example, consider the task of predicting the polarity of a movie review
- The prediction clearly does not depend only on the last word but also on some words which appear before
- Here again we could think that the network is performing the same task at each step (input : word, output : +/−) but it's just that we don't care about intermediate outputs



- Sequences could be composed of anything (not just words)
- For example, a video could be treated as a sequence of images
- We may want to look at the entire sequence and detect the activity being performed