CS725

So far, all neural network families worked with fixed-length inputs and fixed-length outputs

E.g. Image classification. Input $x \in \mathbb{R}^{n \times n}$ or $x' \in \mathbb{R}^d$

Need to support variable-length inputs and outputs

Examples of variable length input/output tasks

(A) Fixed-length i/p & variable-length o/p

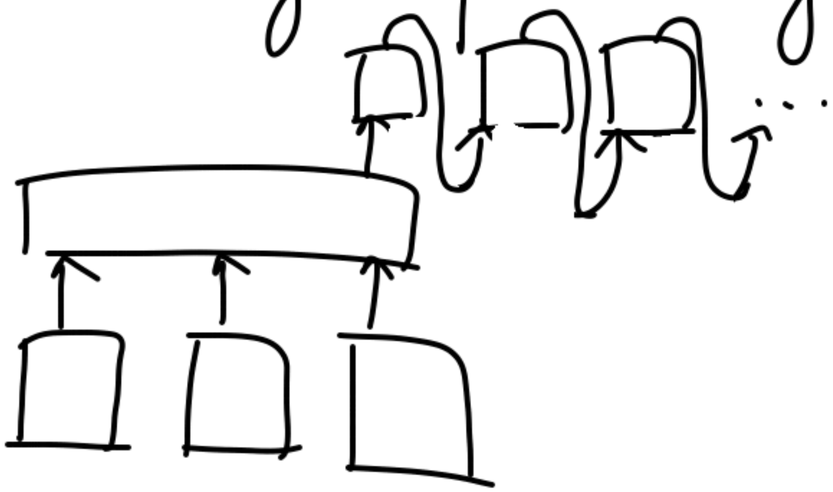
E.g. Image captioning

(B) Variable-length i/p & fixed length o/p

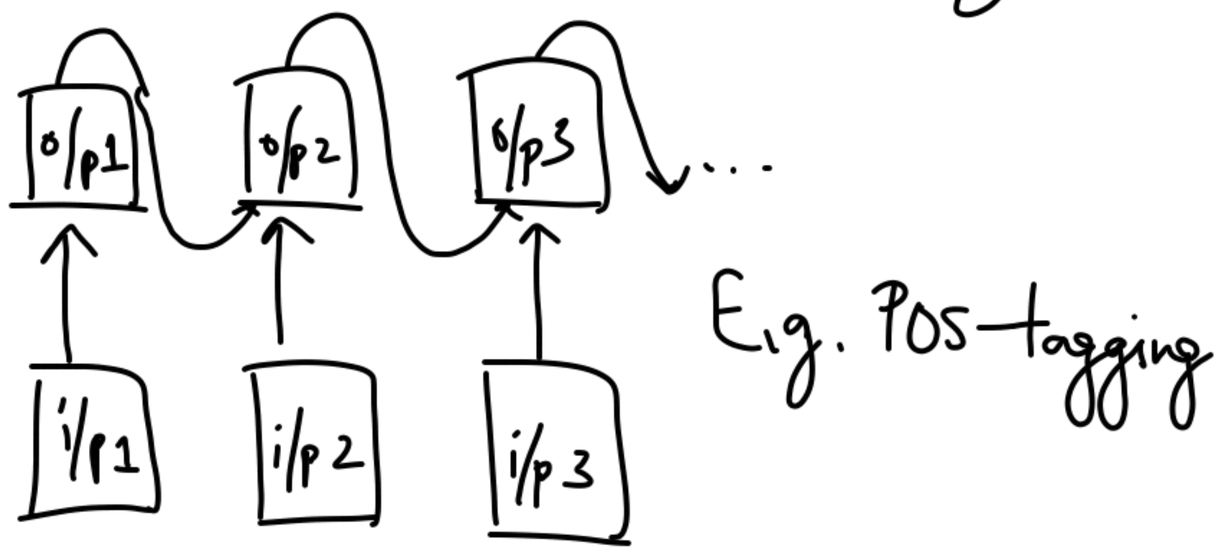
E.g. Image generation, Speech emotion classification, Sentiment analysis, etc
or text

(C) Var-length i/p & var-length outputs

E.g. Speech recognition (ASR), machine translation (MT), Summarization, etc



ASR, MT, Summ, ...



E.g. POS-tagging

Sequence labeling tasks where length of the i/p & o/p seqs are the same

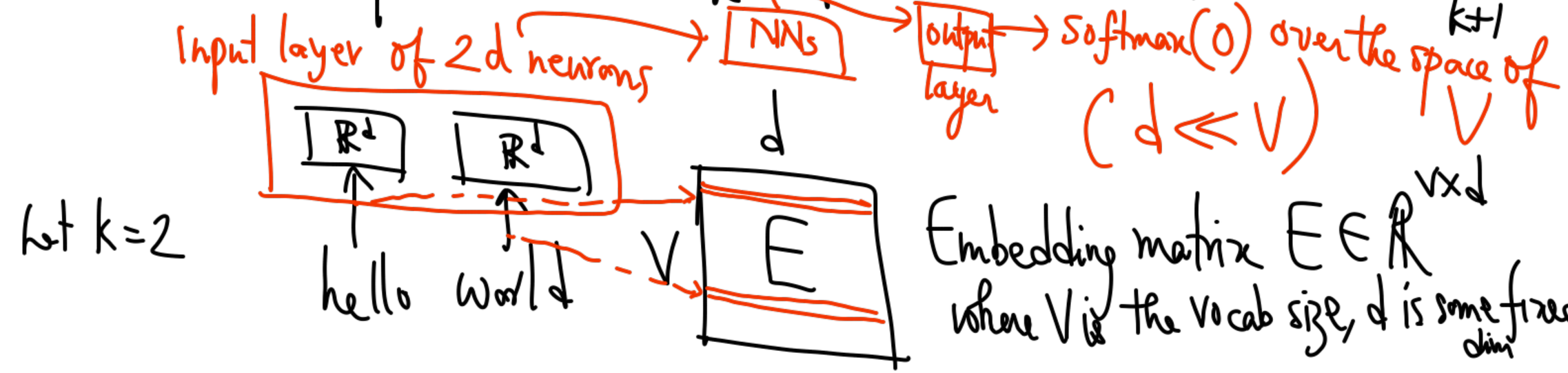
LANGUAGE MODELING task

What is language modeling? Given a sequence of words $w_1, \ldots, w_{T-1}$, what is the most likely next word?

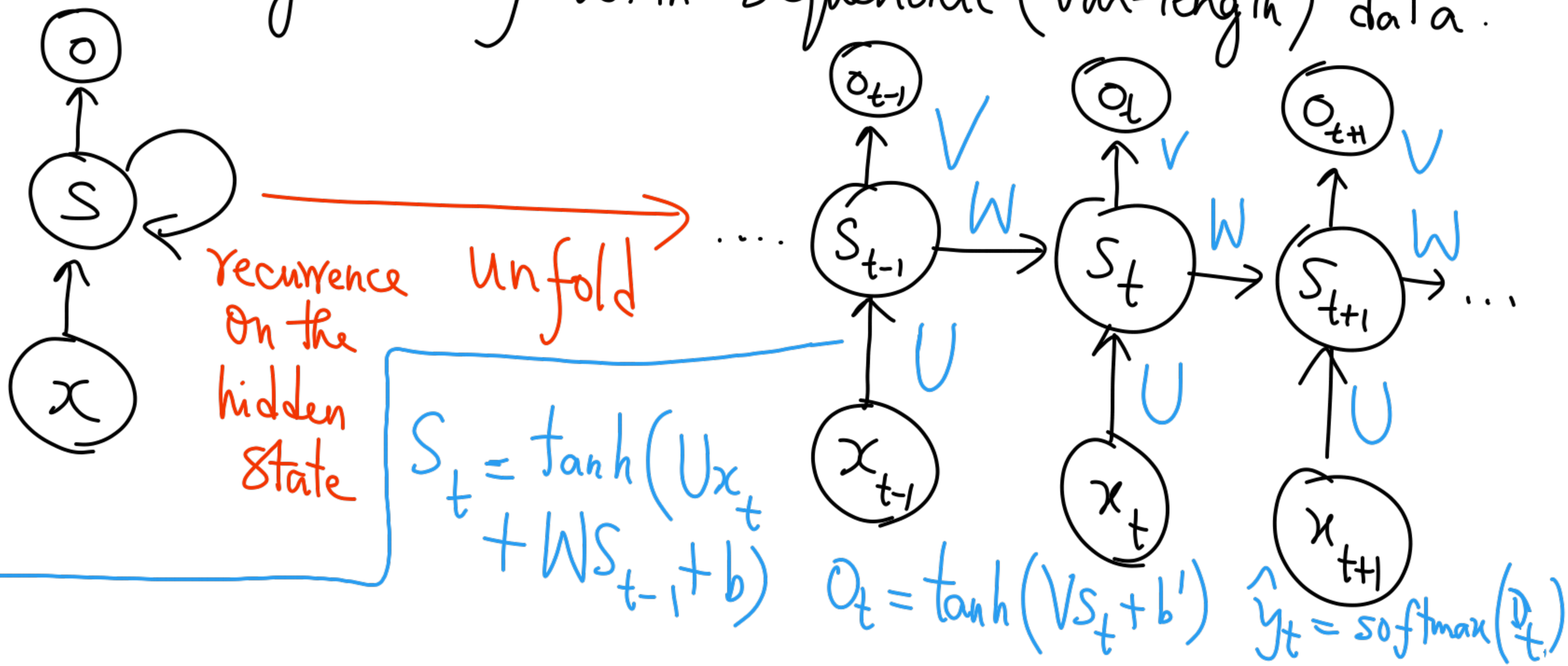
$$w^* = \underset{w}{\operatorname{argmax}}\, P\left(w \middle| w_1, \ldots, w_{T-1}\right)$$

How can we use feedforward neural networks for language modeling?

fix a window size, say k and feed k words at a time as input $w_1 \dots w_k$ to predict the $(k+1)^{th}$ word w_{k+1}

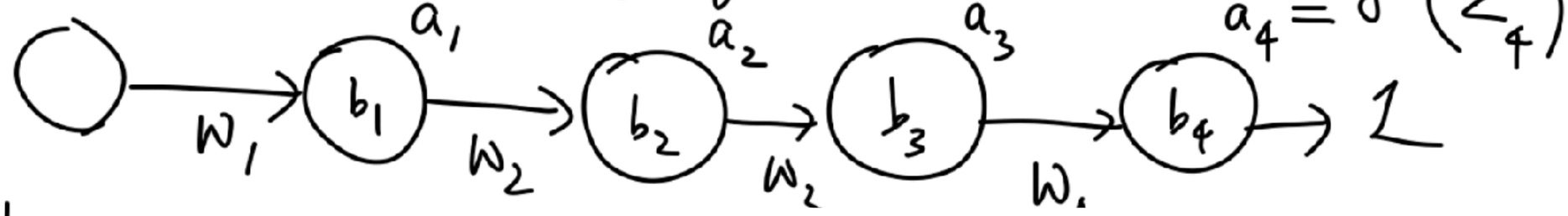


Unlike FFNs, recurrent neural networks (RNNs) deal organically with sequential (van-length) data.



A challenge of RNNs ⇒ Vanishing or exploding gradients

Illustration of vanishing gradient:



$$\frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial a_4} \cdot \frac{\partial a_4}{\partial z_4} \cdot \frac{\partial z_4}{\partial a_3} \cdot \frac{\partial a_3}{\partial z_3} \cdot \frac{\partial z_3}{\partial a_2} \cdot \frac{\partial a_2}{\partial z_2} \cdot \frac{\partial z_2}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial b_1}$$

$$= \frac{\partial L}{\partial a_4} \cdot \sigma'(z_4) \cdot w_4 \cdot \sigma'(z_3) \cdot w_3 \cdot \sigma'(z_2) \cdot w_2 \cdot \sigma'(z_1)$$

Gradients become smaller in early layers (if wts are typically < 1)

Vanishing gradients when we consider gradients from early layers and the wts are initialized to values < 1

Exploding gradients can happen when you start with large weights

Clipping threshold to keep exploding gradients in check!