

# practical deep learning

## recurrent neural networks

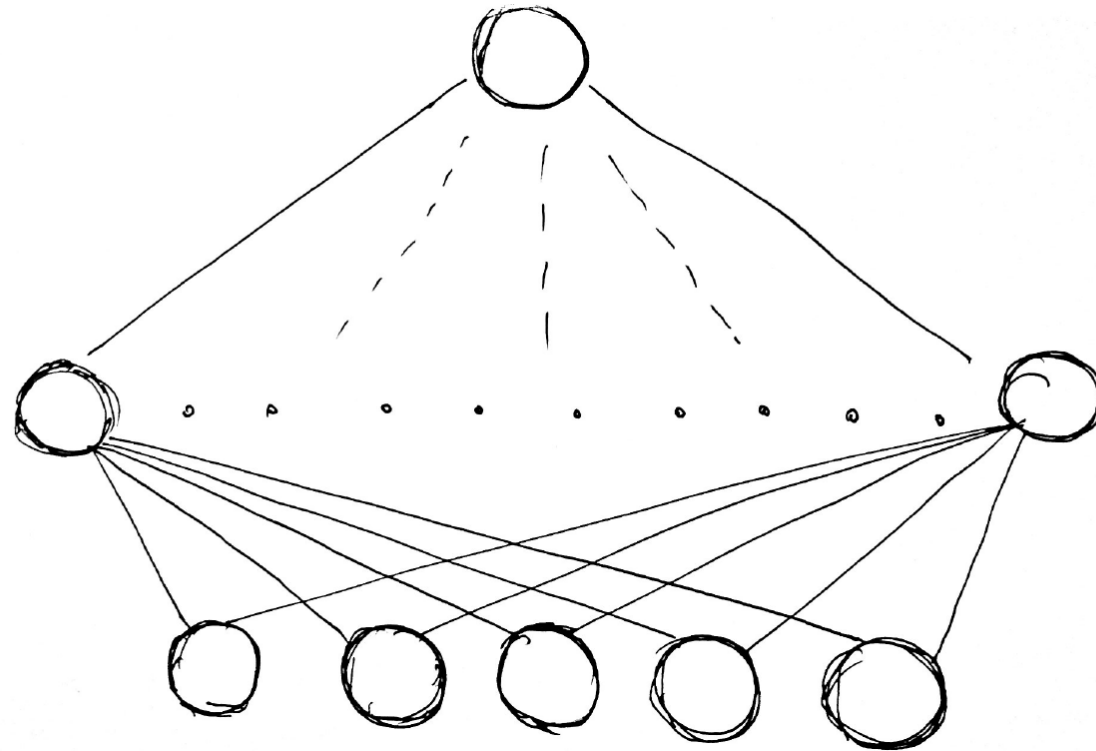
Alex Honchar  
University of Verona

# Day 4 goals

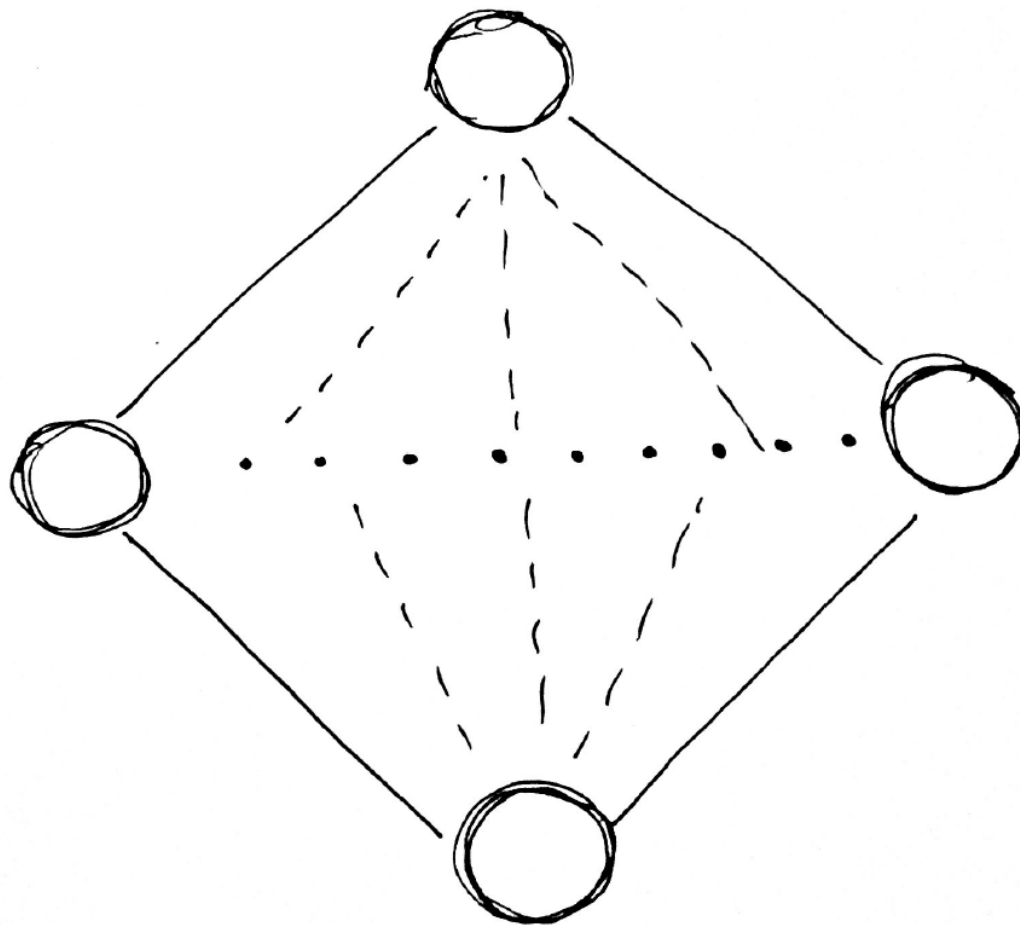
- You **understand** how recurrent neural network (RNN) works
- You **can** train your own RNN for text analysis
- You **can** use word2vec for text analysis
- You **can** train your own RNN for time series analysis
- You **can** use **TensorFlow, Keras (again) and gensim** frameworks

RNN

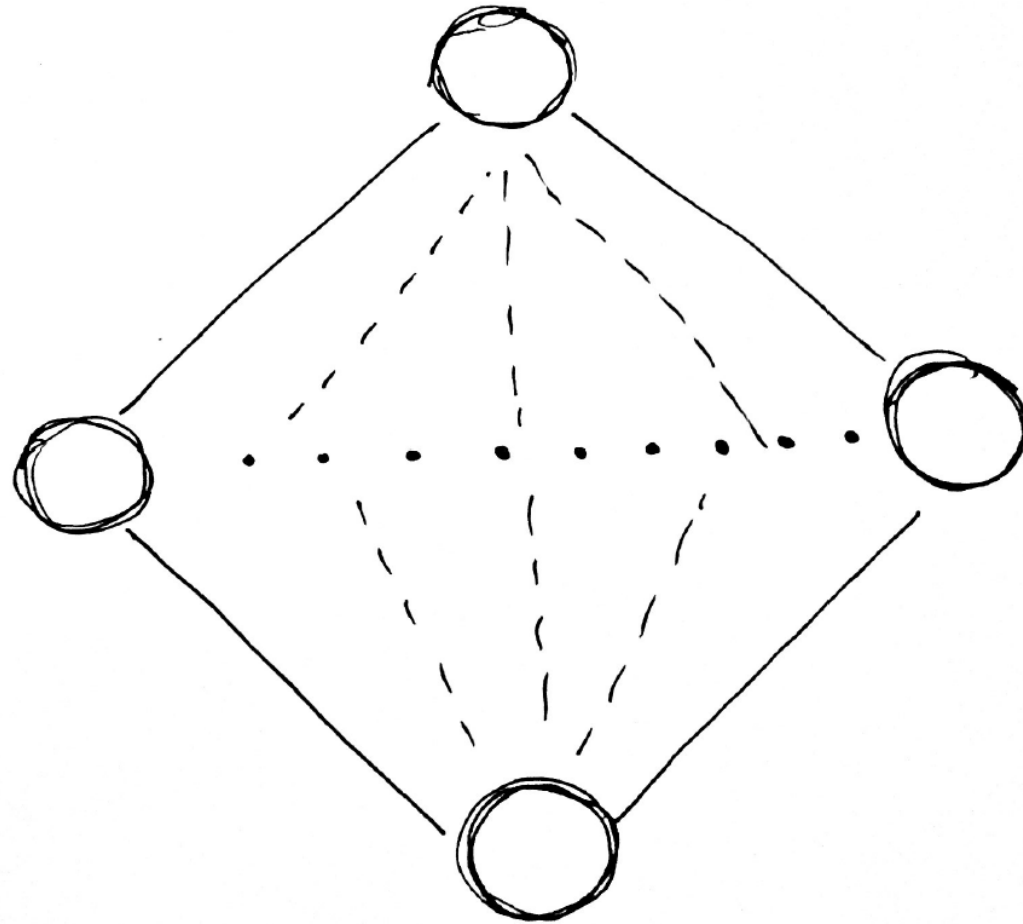
# Multilayer perceptron



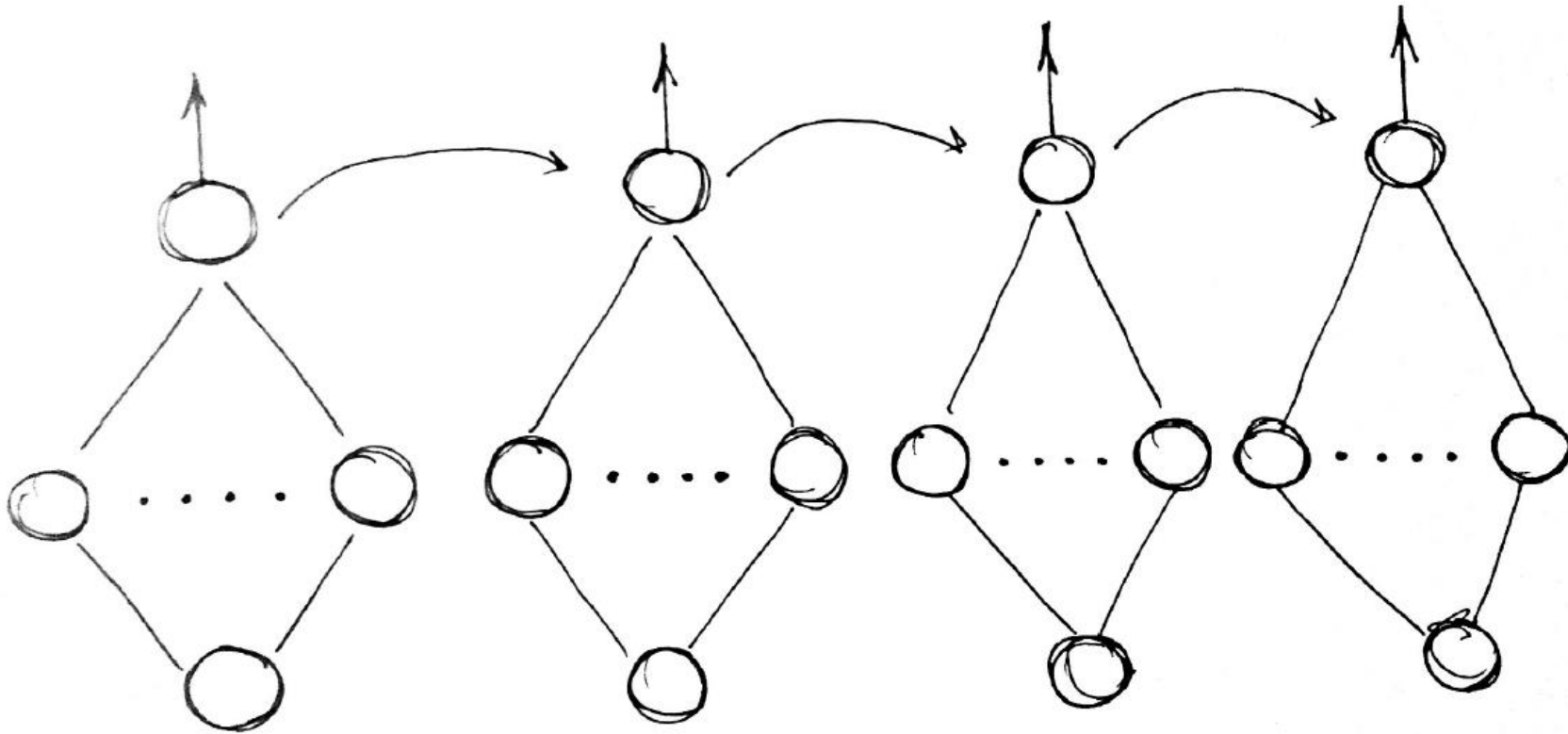
# Single input to MLP



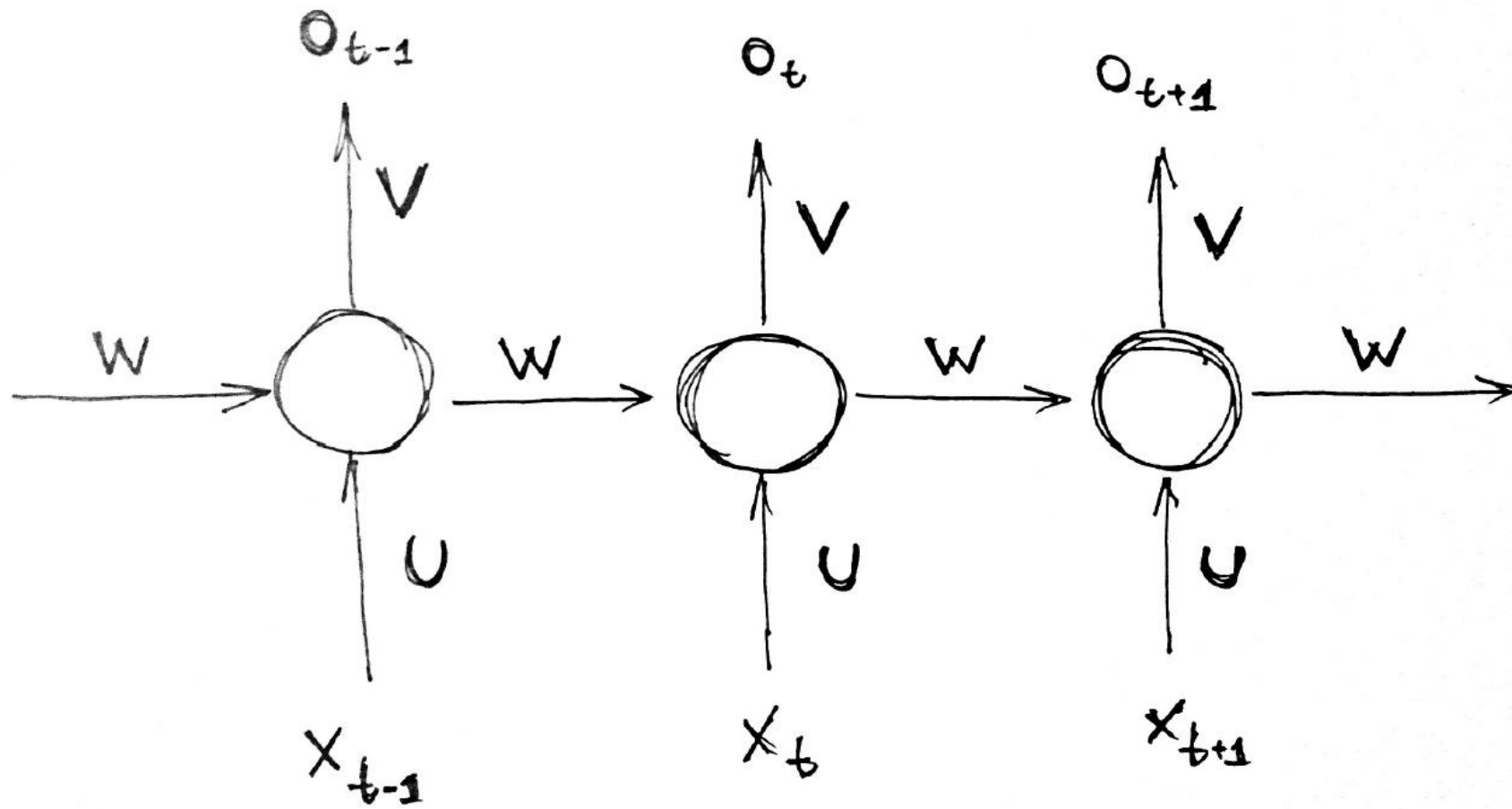
Loop it!



# Recurrent neural network



# Weights in RNN





Hidden state and output

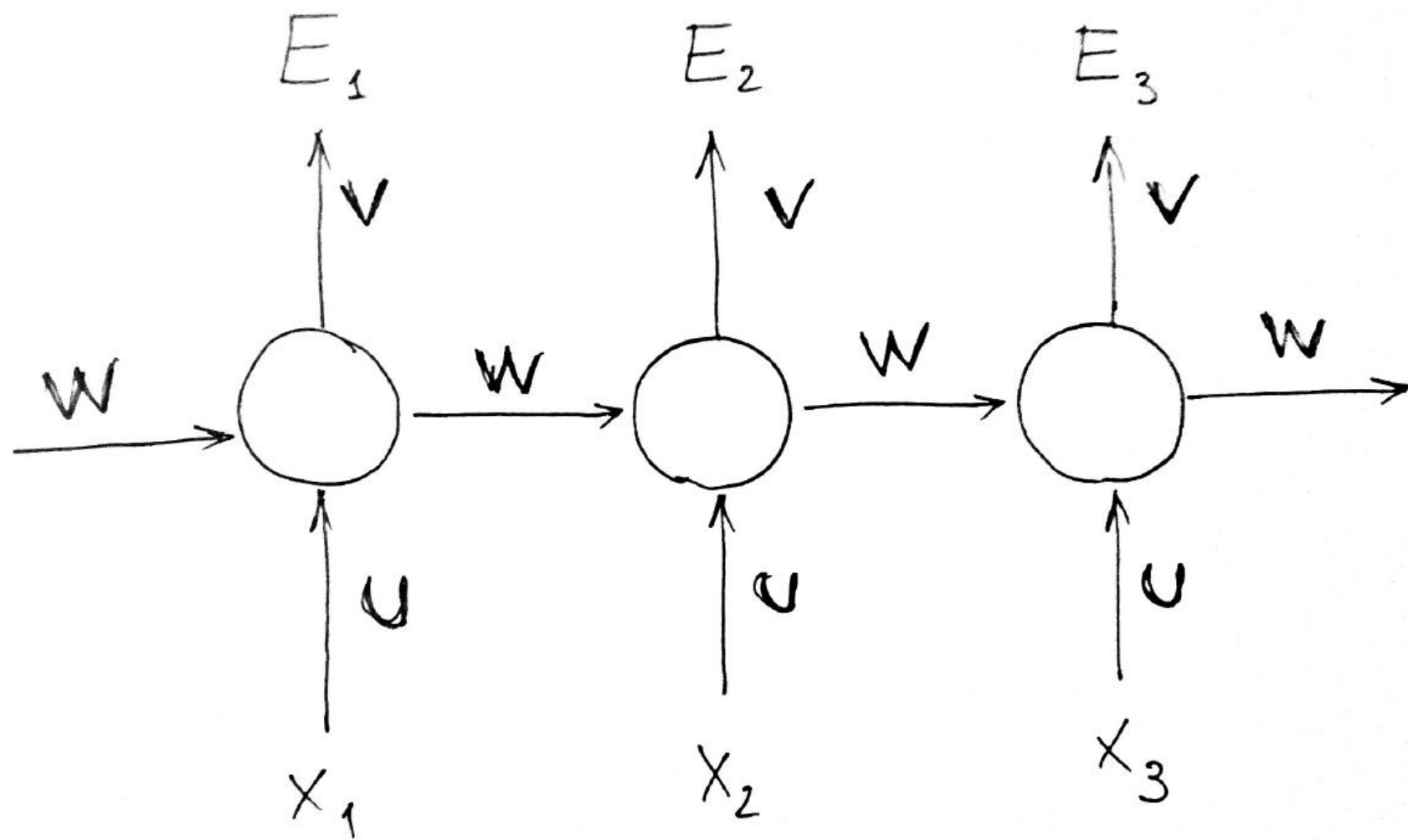
$$S_t = \sigma(Ux_t + Ws_{t-1})$$

$$O_t = \sigma(Vs_t)$$

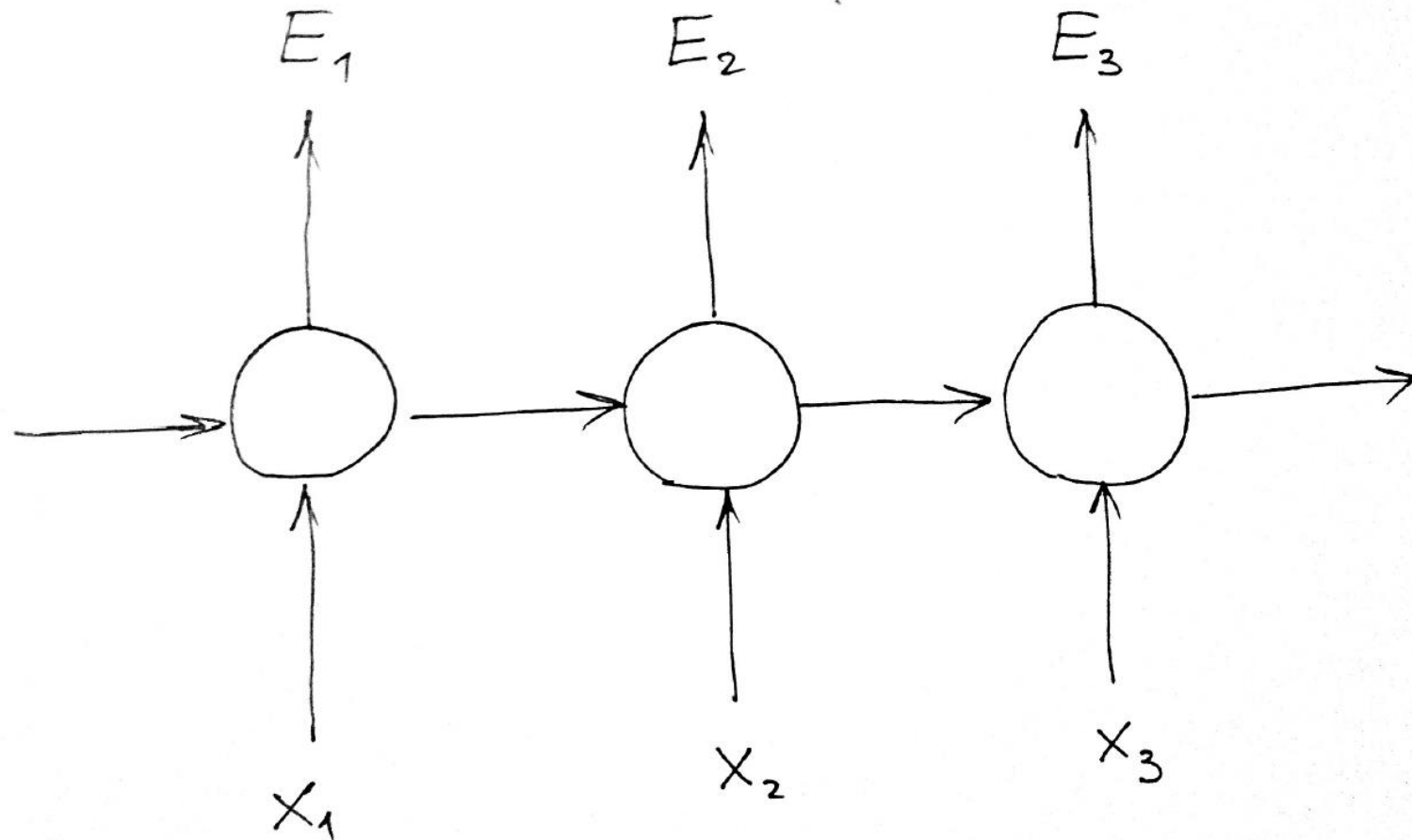
Error = sum of errors

$$E_t(y_t, \hat{y}_t) = -y_t \log \hat{y}_t$$

$$E(y, \hat{y}) = \sum_t E_t(y_t, \hat{y}_t)$$



# Backpropagation through time



# Backpropagation through time

$$\frac{\partial \bar{E}_t}{\partial V} = \frac{\partial \bar{E}_t}{\partial O_t} \frac{\partial O_t}{\partial V} = \frac{\partial E}{\partial O_t} \frac{\partial O_t}{\partial z_t} \frac{\partial z_t}{\partial V},$$

$$z_t = V S_t$$

# Backpropagation through time

$$\begin{aligned}\frac{\partial E_t}{\partial W} &= \frac{\partial E_t}{\partial O_t} \frac{\partial O_t}{\partial W} = \frac{\partial E_t}{\partial O_t} \frac{\partial O_t}{\partial S_t} \frac{\partial S_t}{\partial W} = \\ &= \frac{\partial E_t}{\partial O_t} \frac{\partial O_t}{\partial S_t} \frac{\partial S_t}{\partial S_{t-1}} \frac{\partial S_{t-1}}{\partial W} = \\ &= \frac{\partial E_t}{\partial O_t} \frac{\partial O_t}{\partial S_t} \frac{\partial S_t}{\partial S_{t-1}} \frac{\partial S_{t-1}}{\partial S_{t-2}} \frac{\partial S_{t-2}}{\partial W} = \\ &= \dots\end{aligned}$$

# Backpropagation through time

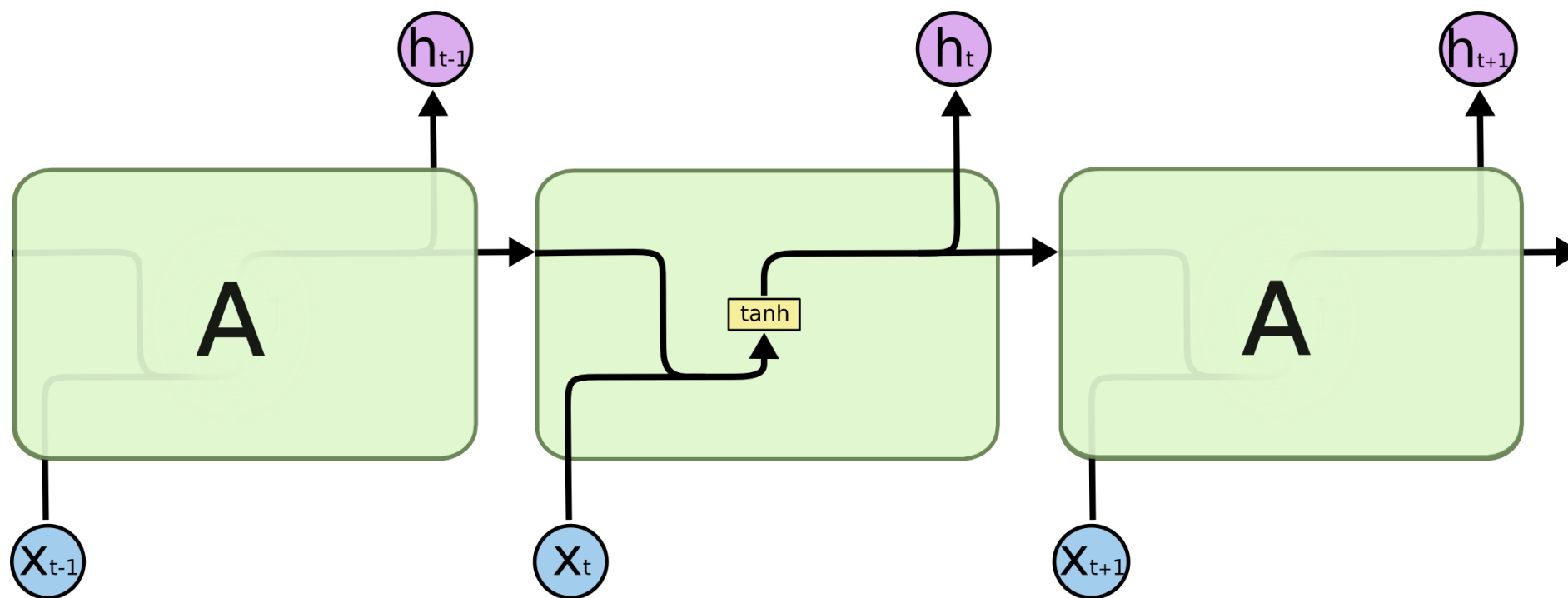
$$\frac{\partial E_t}{\partial U} = \frac{\partial E_t}{\partial O_t} \frac{\partial O_t}{\partial U} = \frac{\partial E_t}{\partial O_t} \frac{\partial O_t}{\partial S_t} \frac{\partial S_t}{\partial U}$$

# long-short term memory

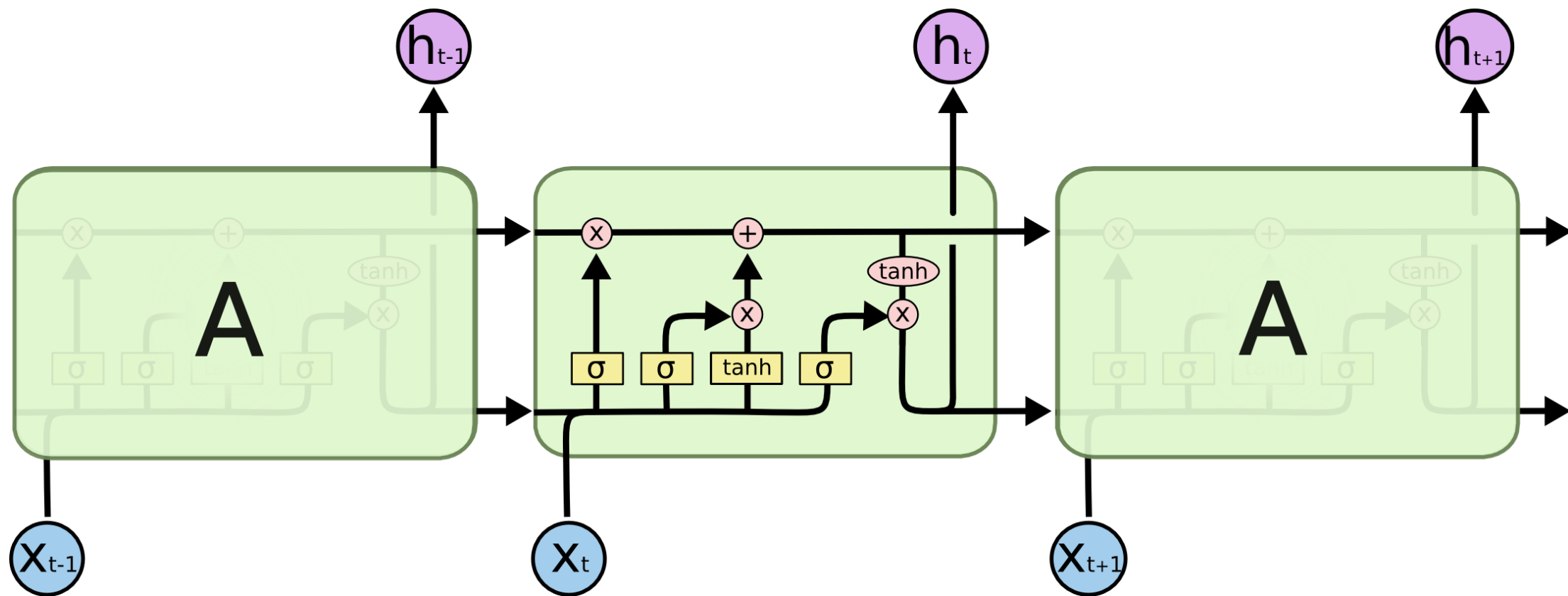
<http://colah.github.io/posts/2015-08-Understanding-LSTMs/>



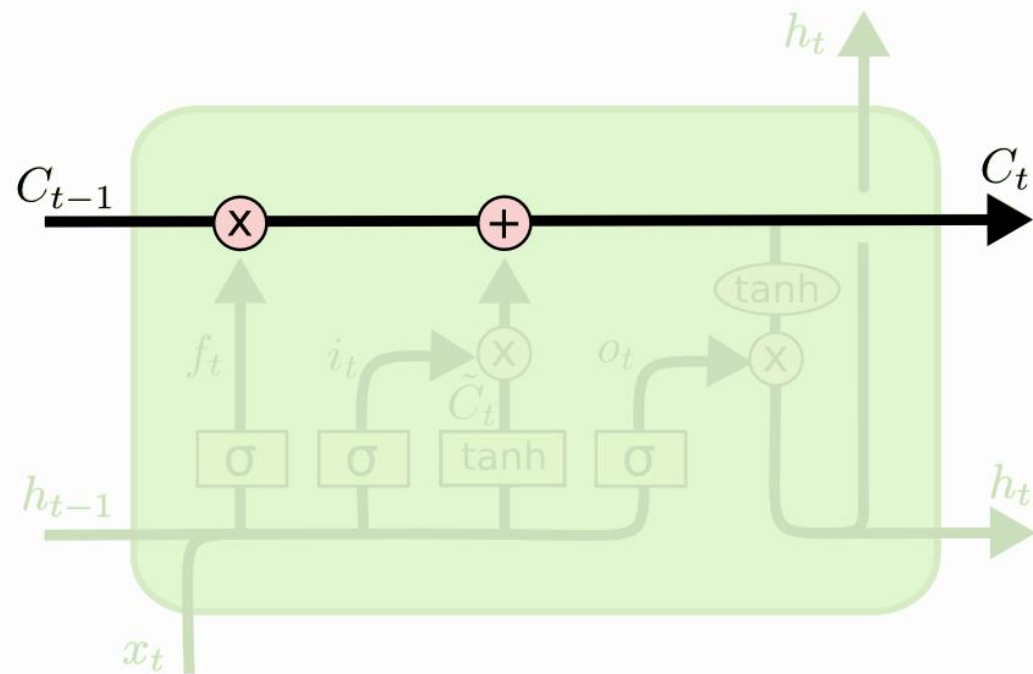
# RNN



# LSTM



# LSTM



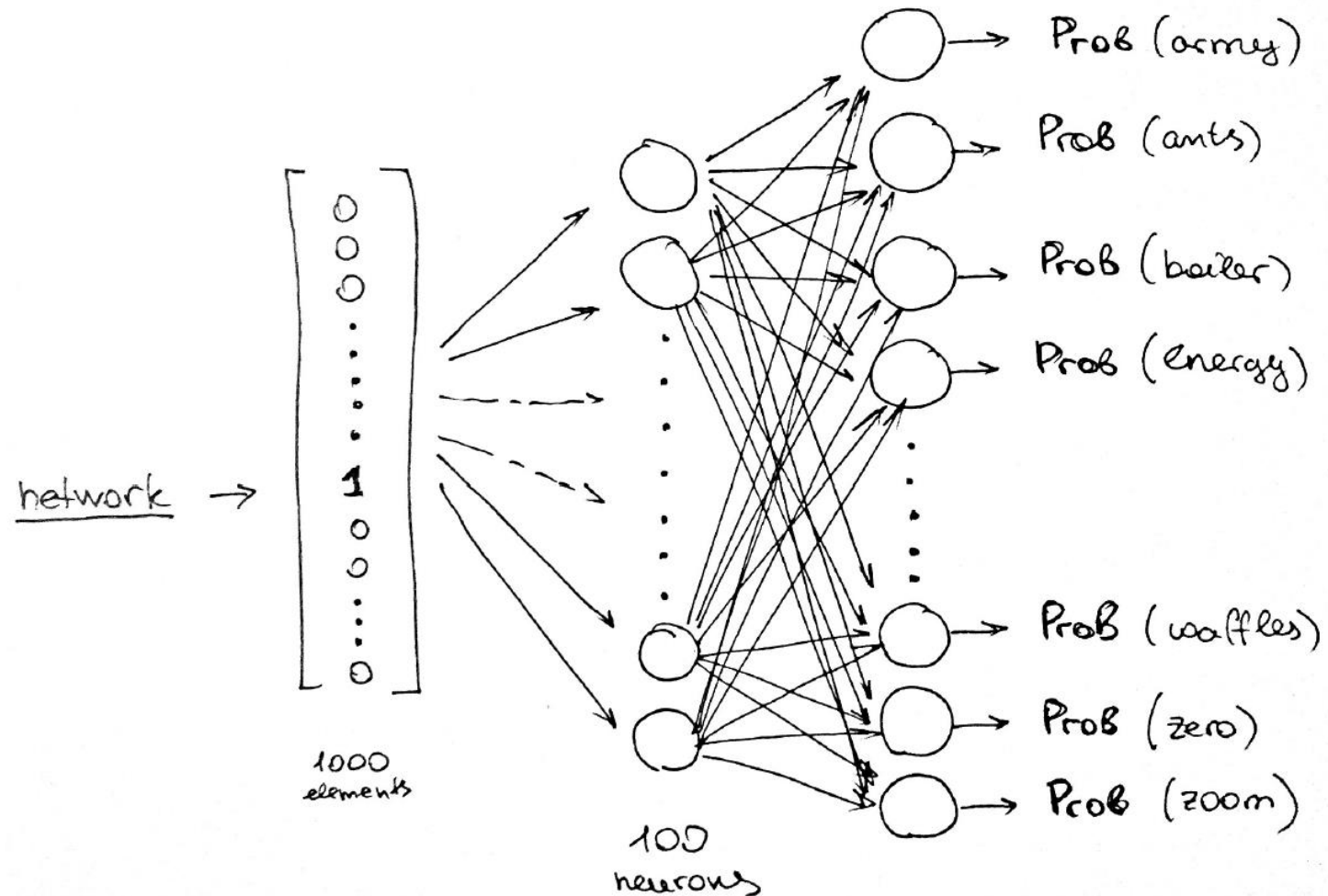
word2vec

# Vocabulary

1000 words

army	$[1, 0, 0, \dots, 0, 0]$	1000 elements
ants	$[0, 1, 0, \dots, 0, 0]$	1000 elements
.	$[0, 0, 1, \dots, \dots]$	1000 elements
.		
.		
.		
.		
.	$[0, 0, 0, \dots, \dots, 0, 1, 0]$	1000 elements
zoom	$[0, 0, 0, \dots, \dots, 0, 0, 1]$	1000 elements

# Skip-gram model

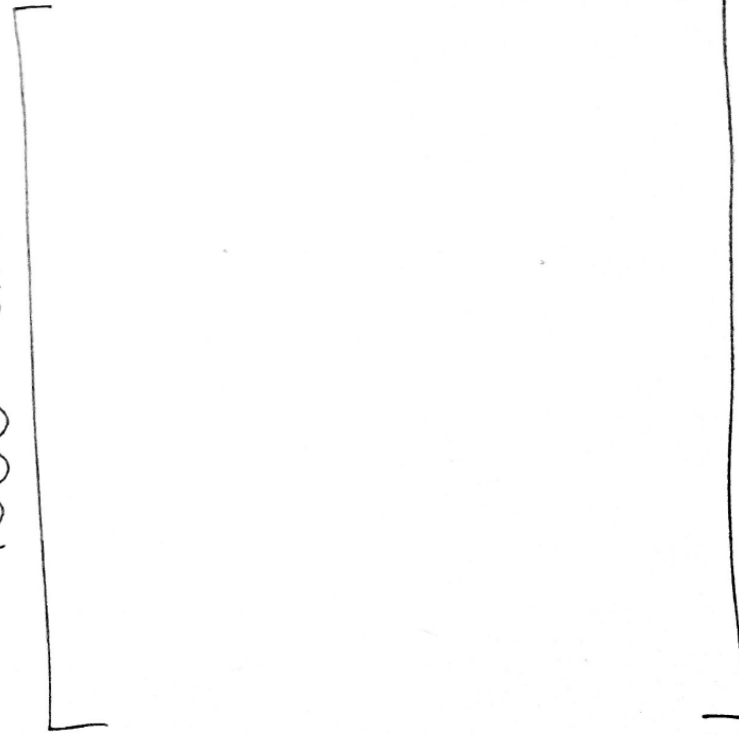


# Weights matrix $\rightarrow$ lookup table

W

100 neurons

1000 words



hidden dynamics in finance



