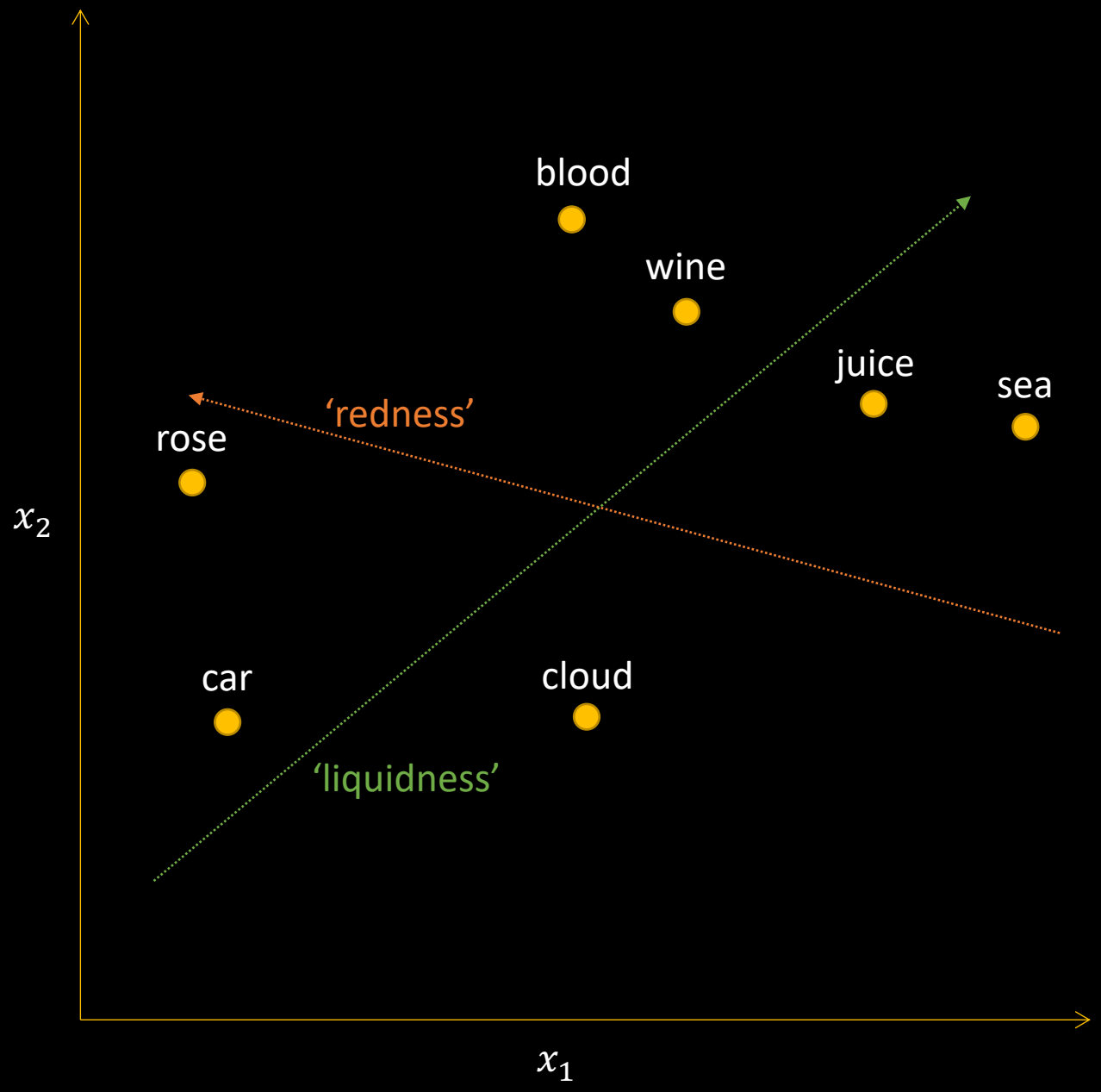
Word Vectors

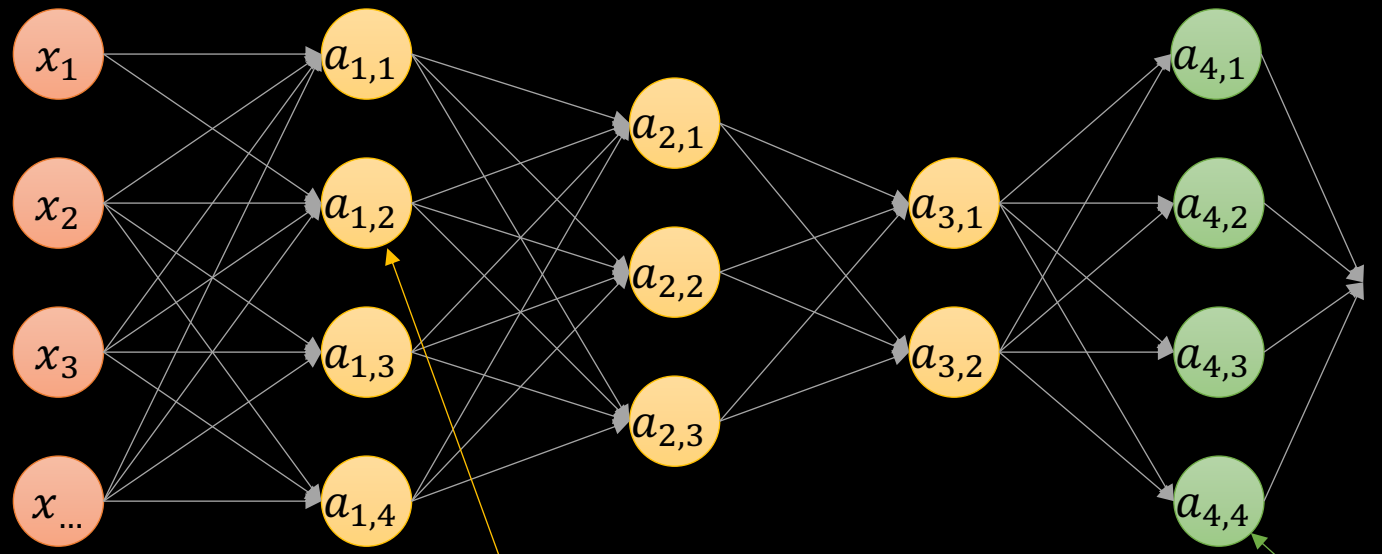


$$rose = \begin{pmatrix} 1.2 \\ 6.9 \end{pmatrix} = \mathbb{R}_2$$

$word$	x_1	x_2
rose	1.2	6.9
blood	5.1	9.3
car	1.5	3.9
sea	9.3	7.1
wine	6.0	8.5
juice	7.9	7.2
cloud	4.9	4.0

Can we predict the next word in a sentence?

‘cat’



$$\sigma(z) = \begin{matrix} 0.1 \\ 0.522 \\ 0.178 \\ 0.2 \end{matrix}$$

Input layer

Hidden layers

Output layer

[0, 0, 1, 0, 0, 0, ….]

input == a ‘one-hot’ vector
‘cat’ is word number 3 in the dictionary

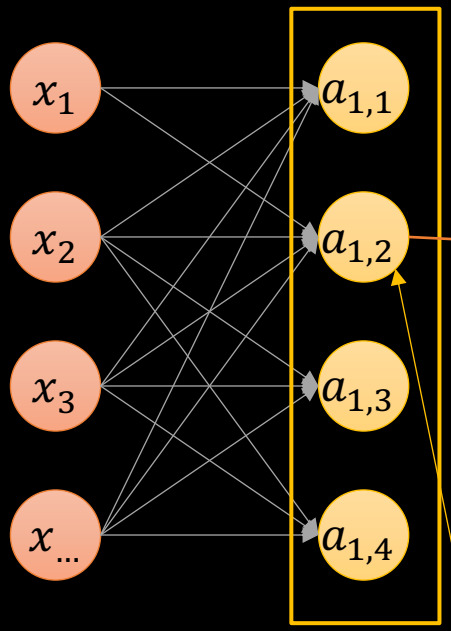
$$a_{1,2} = w_1 \times x_1 + w_2 \times x_2 + w_3 \times x_3 + w_4 \times x_4 + \ \ldots + b$$

each neuron == a linear function of its inputs
one ‘w’ for each word in the dictionary

one output neuron per word in the dictionary

‘cat’ == word 4

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



$$\begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & w_{1,4} \\ w_{2,1} & w_{2,2} & w_{2,3} & w_{2,4} \\ w_{3,1} & w_{3,2} & w_{3,3} & w_{3,4} \\ w_{4,1} & w_{4,2} & w_{4,3} & w_{4,4} \end{bmatrix}$$

This set of numbers represents the network’s idea of the word ‘cat’.

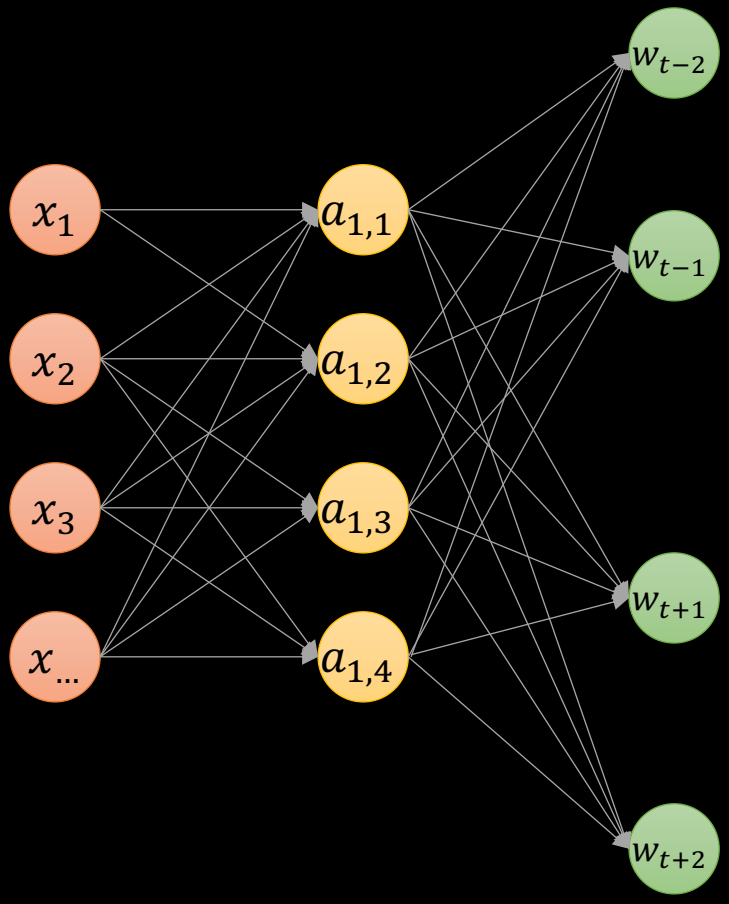
$$a_{1,2} = w_1 \times x_1 + w_2 \times x_2 + w_3 \times x_3 + w_4 \times x_4 + \ \ldots + b$$

each neuron == a linear function of its inputs

one ‘w’ for each word in the dictionary

The ‘word2vec’ skip-gram model

‘cat’



‘a’

‘black’

‘walked’

‘by’

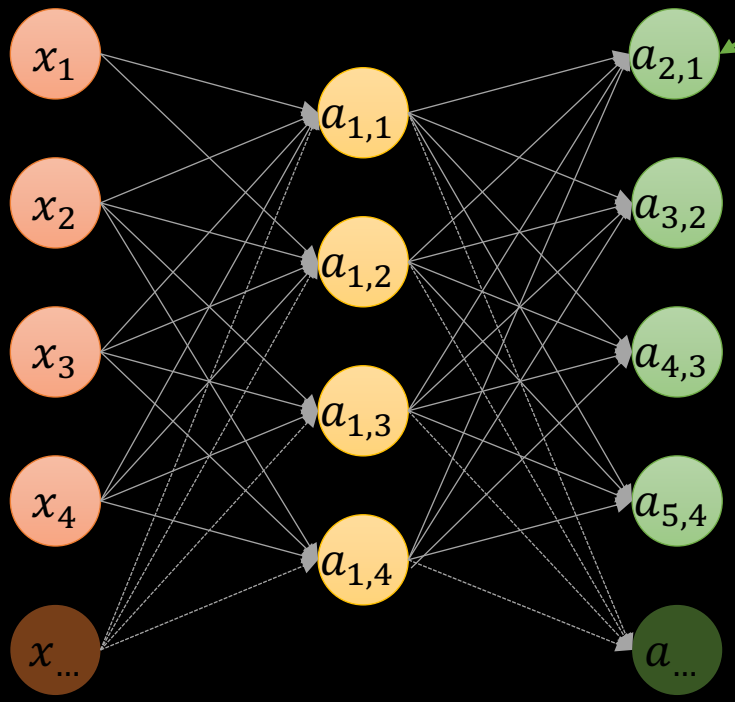
Given this word …

… predict which words appear nearby (window_size == 2)

‘word2vec’ with negative sampling

One prediction neuron for each word in vocabulary (10,000 words = 10,000 neurons)

‘cat’



‘budgie’	0
‘black’	1
‘marmite’	0
‘iPhone’	0
‘...’	...

Given this input word …

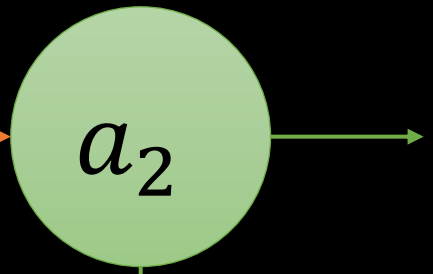
… does the target word appear in the input word’s context?

‘cat’ == word 4

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

$$\begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & w_{1,4} \\ w_{2,1} & w_{2,2} & w_{2,3} & w_{2,4} \\ w_{3,1} & w_{3,2} & w_{3,3} & w_{3,4} \\ w_{4,1} & w_{4,2} & w_{4,3} & w_{4,4} \end{bmatrix}$$

Predictor for ‘black’ (word 2)



1 = ‘black’ is in context

0 = ‘black’ isn’t in context

$$a_2 = \sigma(w_1 \times w_{1,4} + w_2 \times w_{2,4} + w_3 \times w_{3,4} + \cdots + w_n \times w_{n,4})$$

Training data for negative sampling

‘When I looked out, a black cat crossed the road, and purred.’

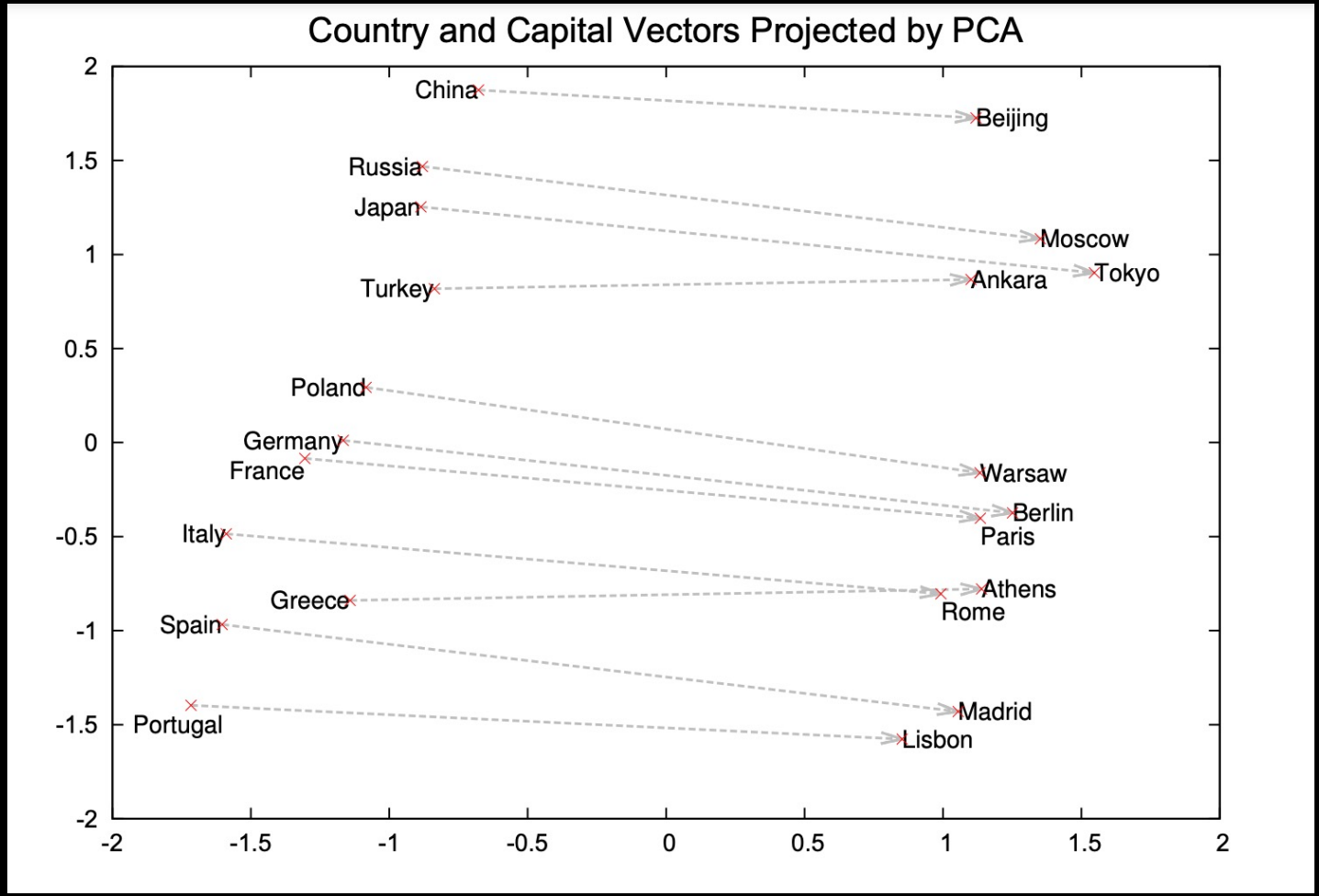
window_size == 3

input word	target word	value
cat	out	1
cat	blue	0
cat	spandex	0
cat	ballet	0
cat	thingamy	0

Actual word

Randomly sampled negative examples

What can you do with word vectors?



Tomas Mikolov and others, 'Distributed Representations of Words and Phrases and Their Compositionality', in *Advances in Neural Information Processing Systems 26*, ed. by C. J. C. Burges and others (Curran Associates, Inc., 2013), pp. 3111–19.