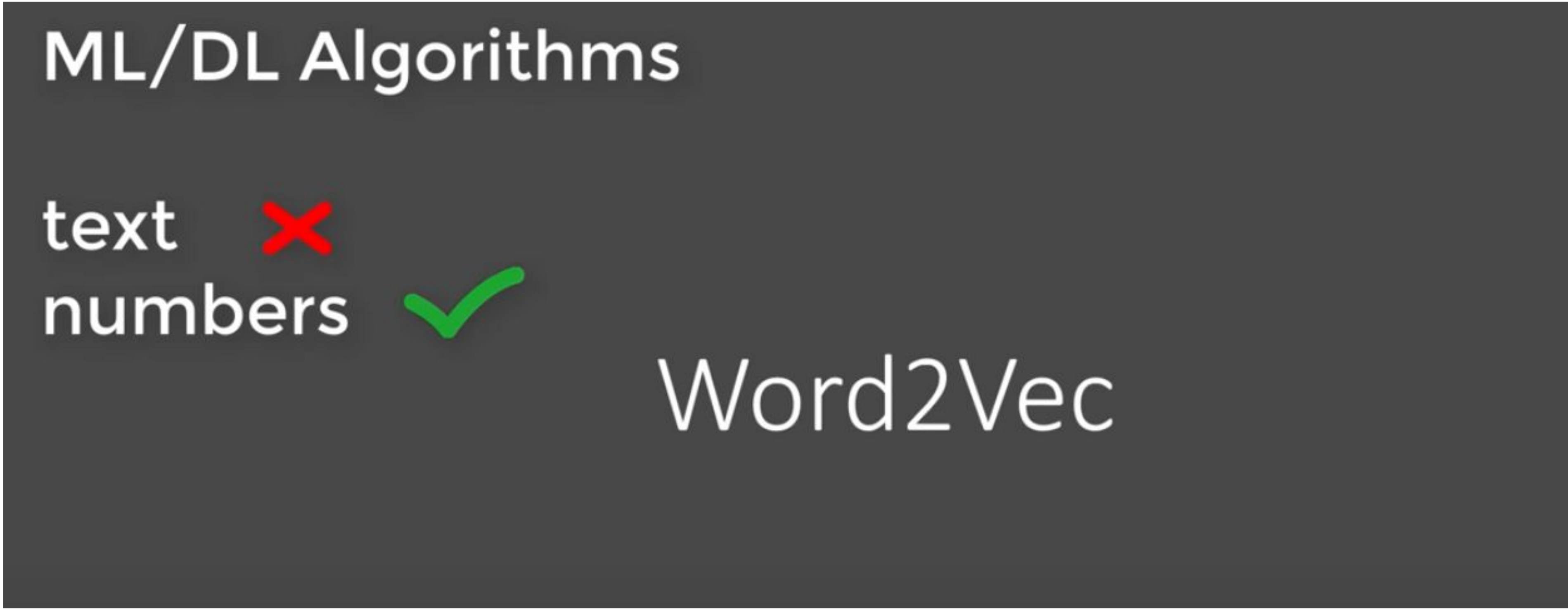
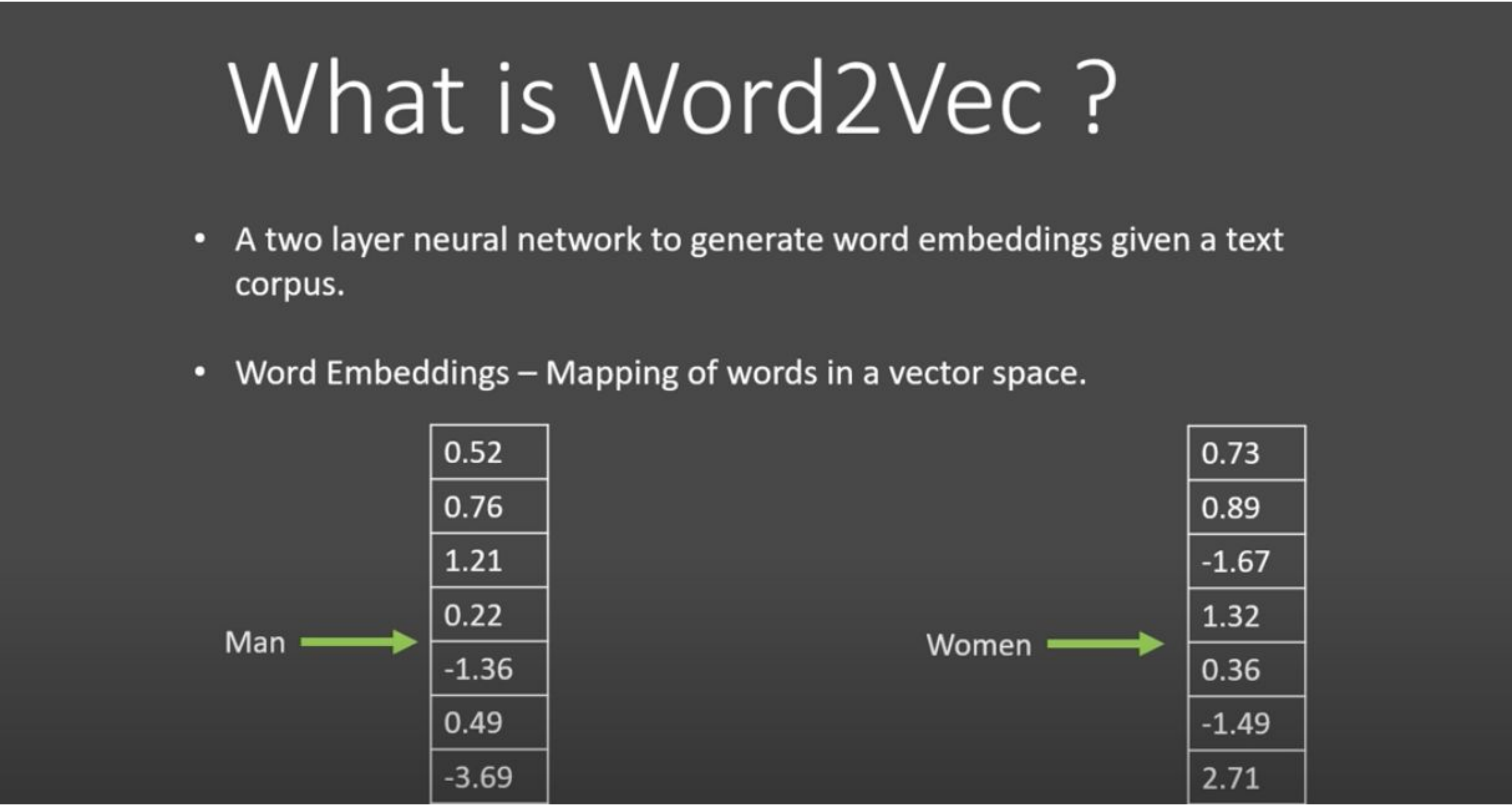


ML/DL Algorithms

text ✗

numbers ✓

Word2Vec



What is Word2Vec ?

- A two layer neural network to generate word embeddings given a text corpus.

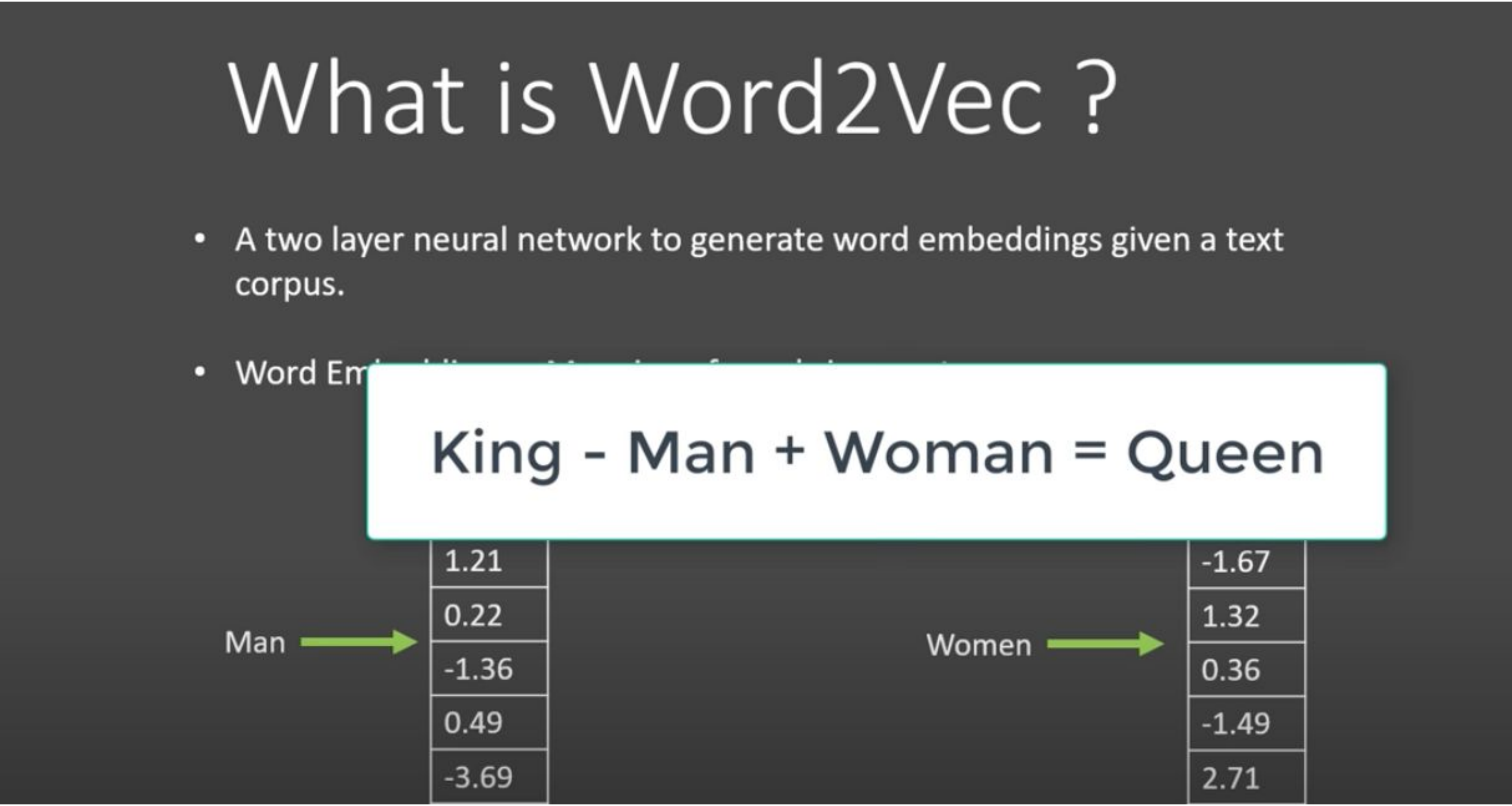
- Word Embeddings – Mapping of words in a vector space.

Man →

0.52
0.76
1.21
0.22
-1.36
0.49
-3.69

Women →

0.73
0.89
-1.67
1.32
0.36
-1.49
2.71



What is Word2Vec ?

- A two layer neural network to generate word embeddings given a text corpus.

- Word Em

King - Man + Woman = Queen

Man

1.21

0.22

-1.36

0.49

-3.69

Women

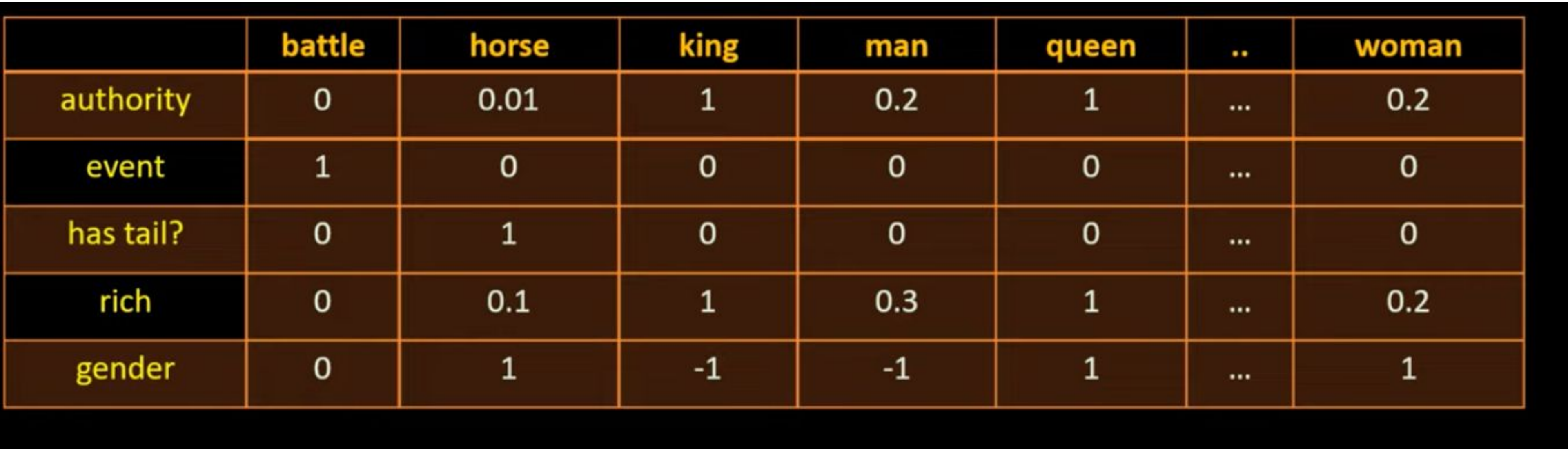
-1.67

1.32

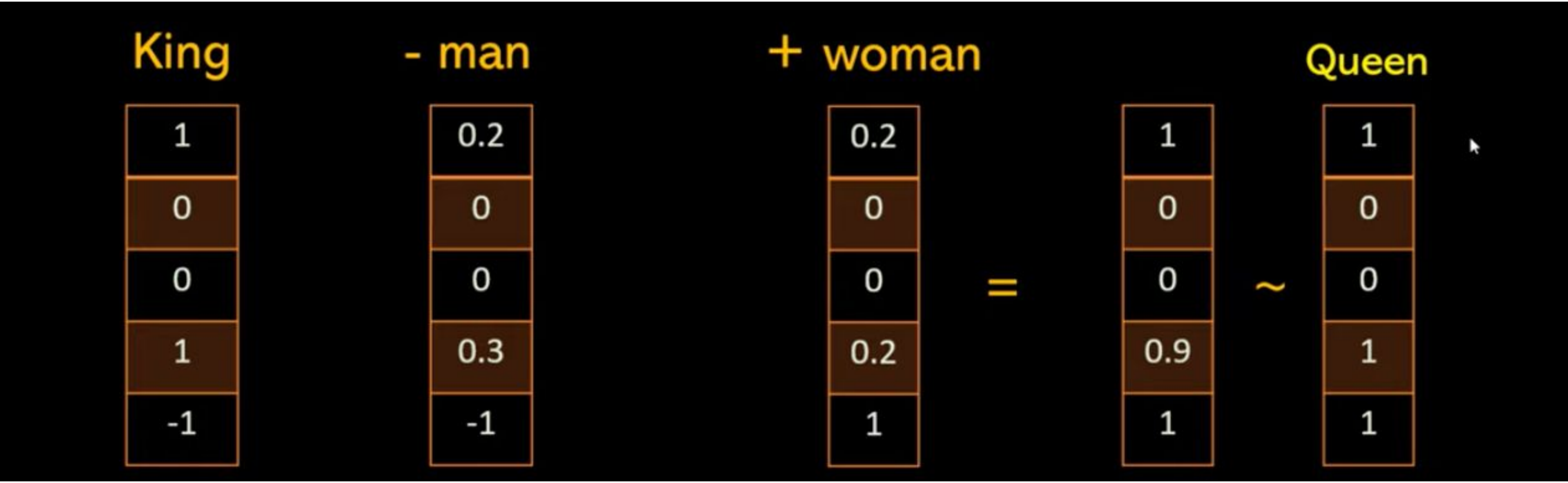
0.36

-1.49

2.71



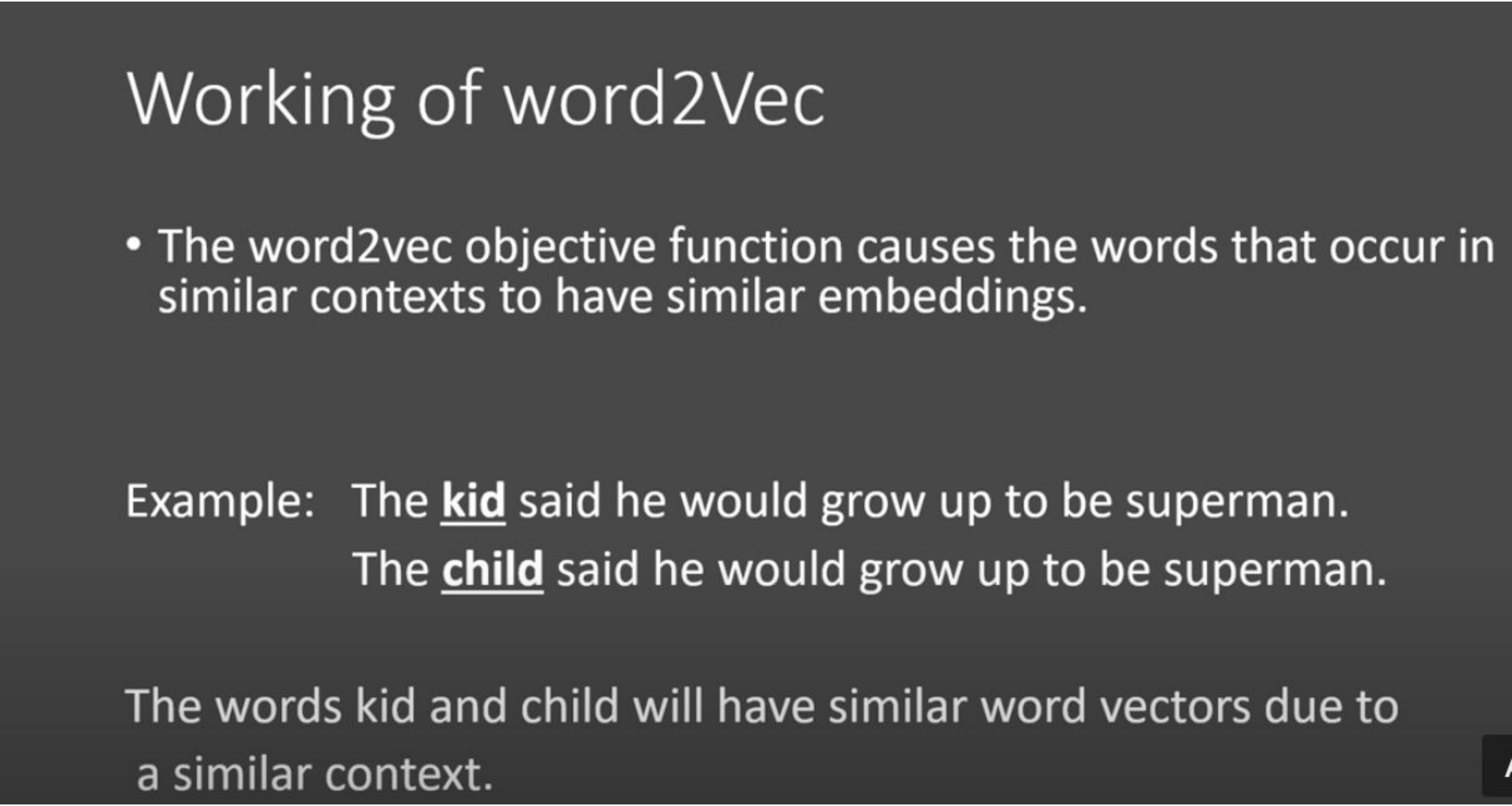
	battle	horse	king	man	queen	..	woman
authority	0	0.01	1	0.2	1	...	0.2
event	1	0	0	0	0	...	0
has tail?	0	1	0	0	0	...	0
rich	0	0.1	1	0.3	1	...	0.2
gender	0	1	-1	-1	1	...	1



King	- man	+ woman				Queen
1	0.2	0.2		1		1
0	0	0		0		0
0	0	0	=	0	~	0
1	0.3	0.2		0.9		1
-1	-1	1		1		1

Why Word2vec?

- Preserves relationship between words.
- Deals with addition of new words in the vocabulary.
- Better results in lots of deep learning applications.



Working of word2Vec

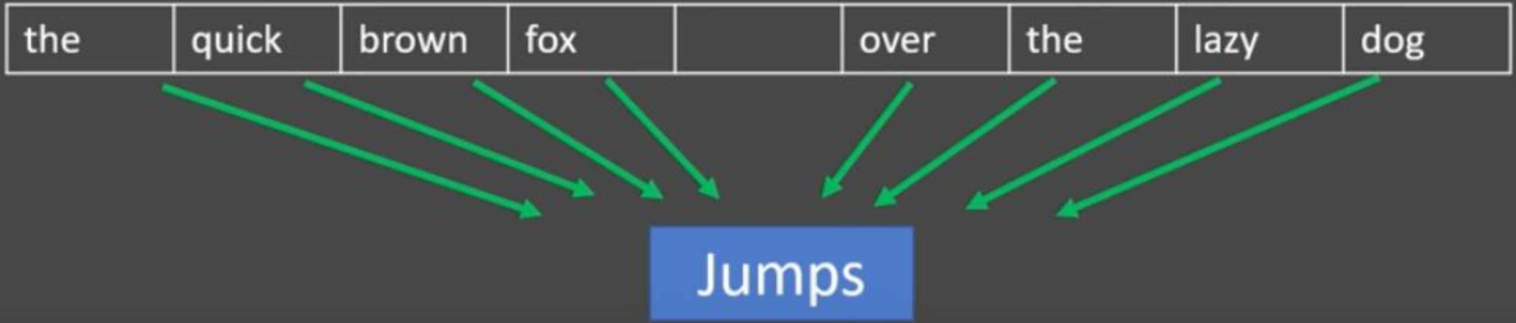
- The word2vec objective function causes the words that occur in similar contexts to have similar embeddings.

Example: The **kid** said he would grow up to be superman.
The **child** said he would grow up to be superman.

The words kid and child will have similar word vectors due to a similar context.

CBOW

- Predict the target word from the context.

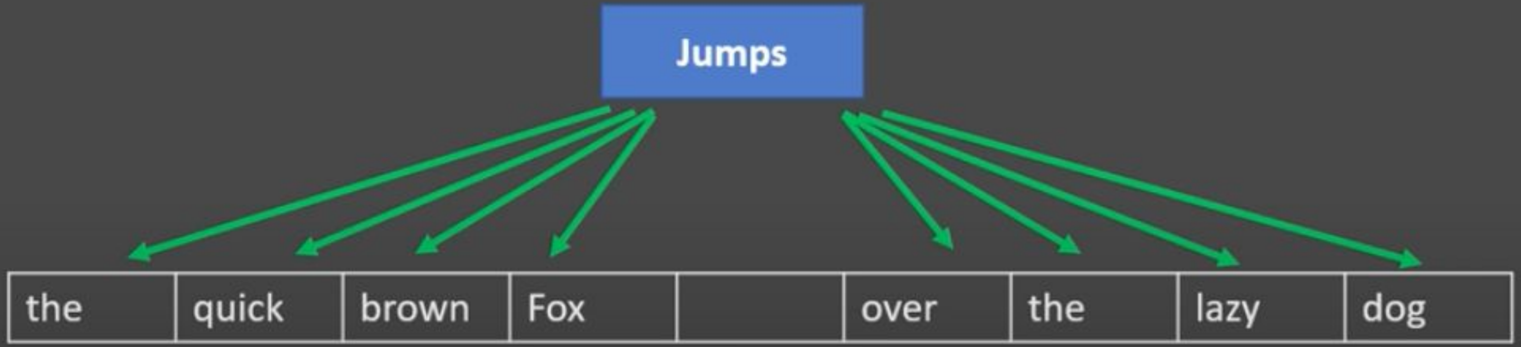


the	quick	brown	fox		over	the	lazy	dog

Jumps

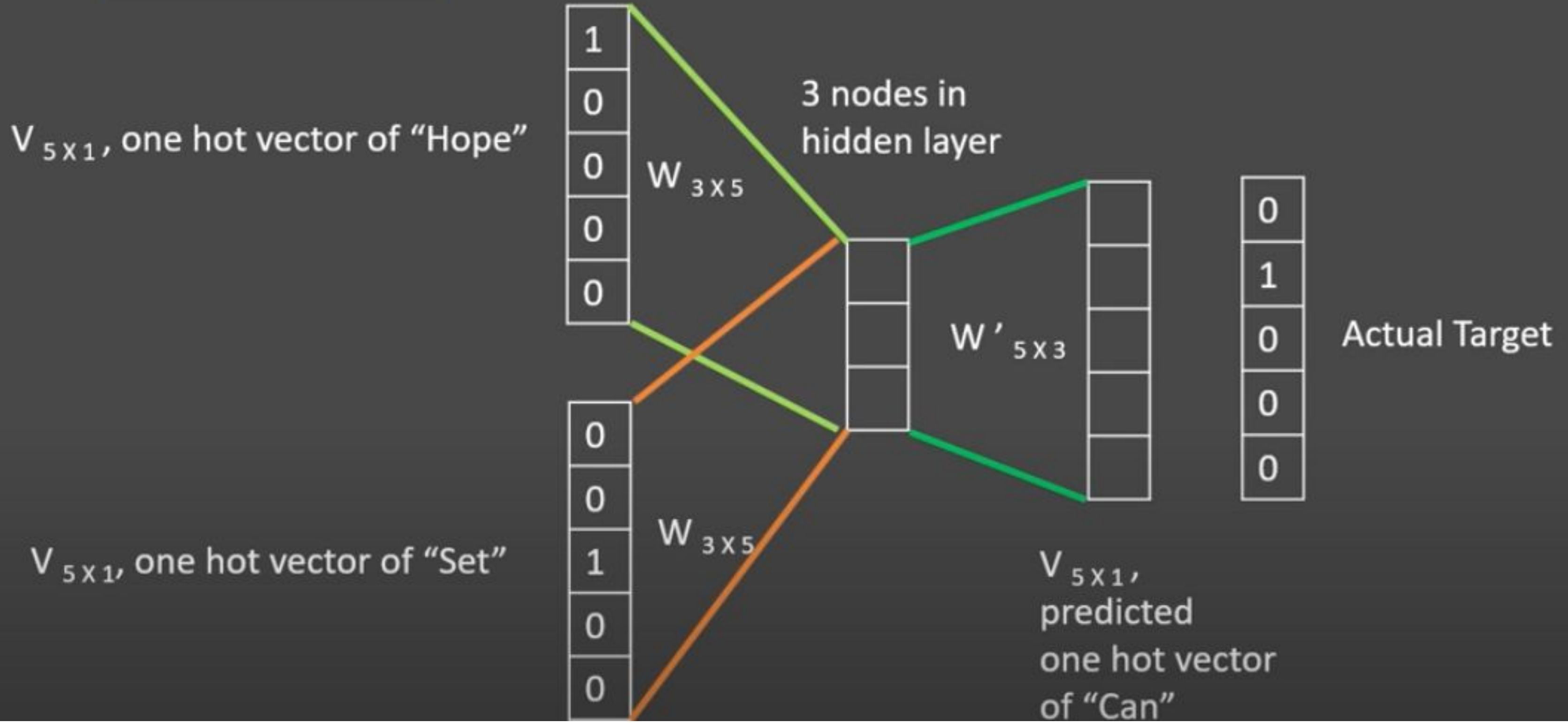
Skip Gram

- Predict the context words from target.



CBOW - Working

Hope can set you free.



V_{5X1}, one hot vector of "Hope"

1 0 0 0 0

W_{3X5}

3 nodes in hidden layer

W'_{5X3}

0 1 0 0 0

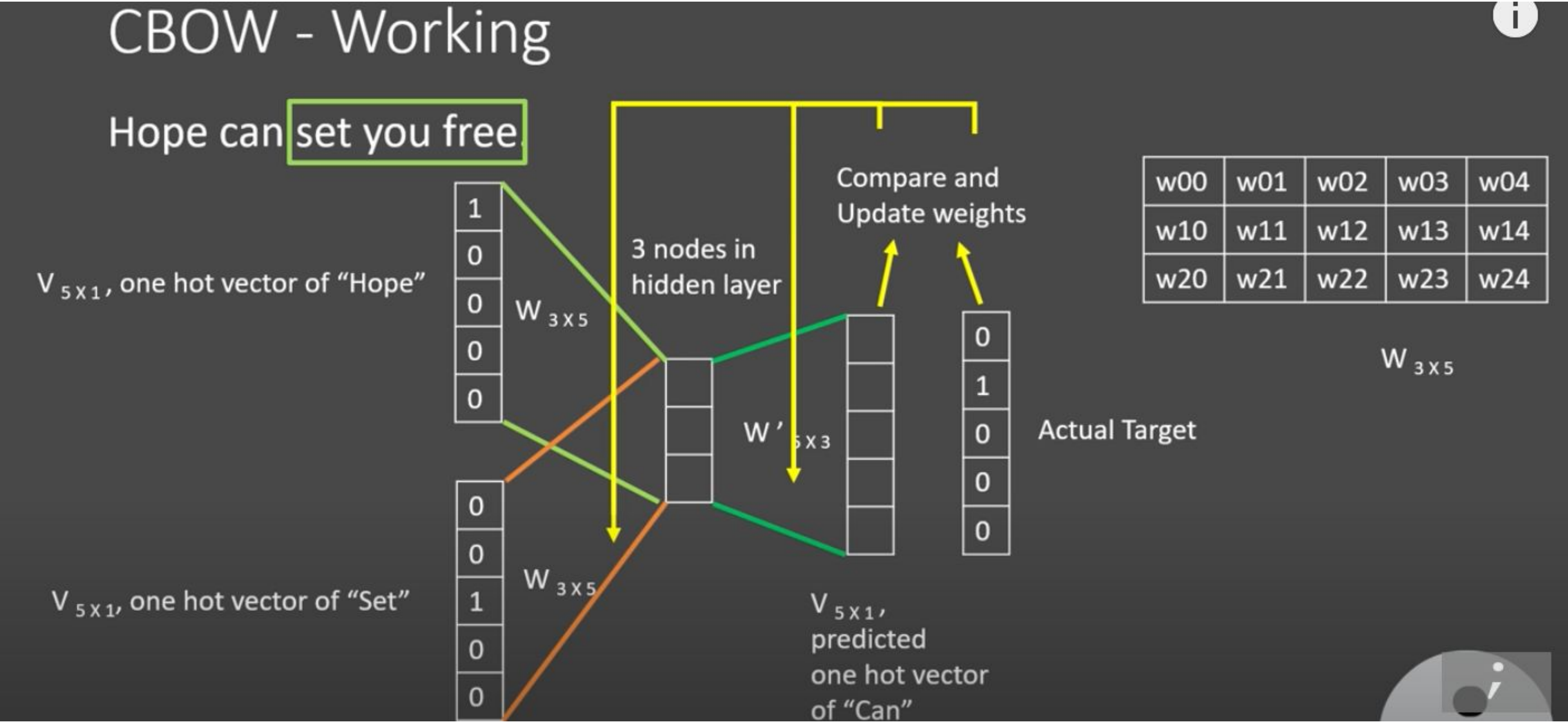
Actual Target

V_{5X1}, one hot vector of "Set"

0 0 1 0 0

W_{3X5}

V_{5X1}, predicted one hot vector of "Can"



CBOW - Working

Hope can set you free.

$V_{5 \times 1}$, one hot vector of "Hope"

1
0
0
0
0

$W_{3 \times 5}$

$V_{5 \times 1}$, one hot vector of "Set"

0
0
1
0
0

$W_{3 \times 5}$

3 nodes in hidden layer

$W'_{5 \times 3}$

Compare and Update weights

$V_{5 \times 1}$, predicted one hot vector of "Can"

0
1
0
0
0

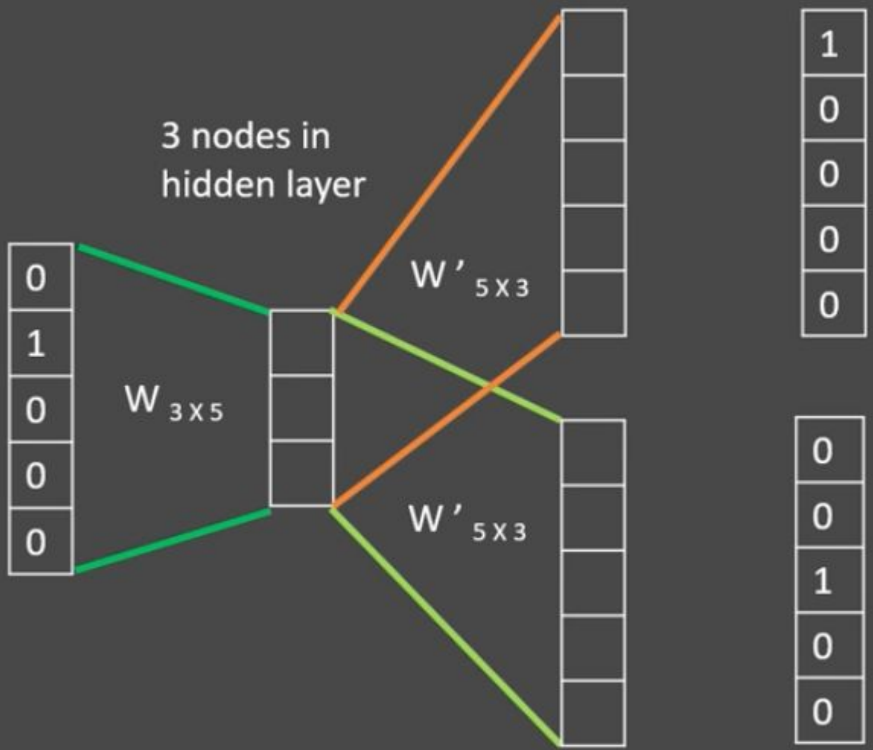
Actual Target

w00	w01	w02	w03	w04
w10	w11	w12	w13	w14
w20	w21	w22	w23	w24

$W_{3 \times 5}$

Skip Gram - Working

Hope can set you free.



$V_{5 \times 1}$, predicted vector of “hope”

Actual Target

1
0
0
0
0

3 nodes in hidden layer

$W'_{5 \times 3}$

$V_{5 \times 1}$, one hot vector of “Can”

0
1
0
0
0

$W_{3 \times 5}$

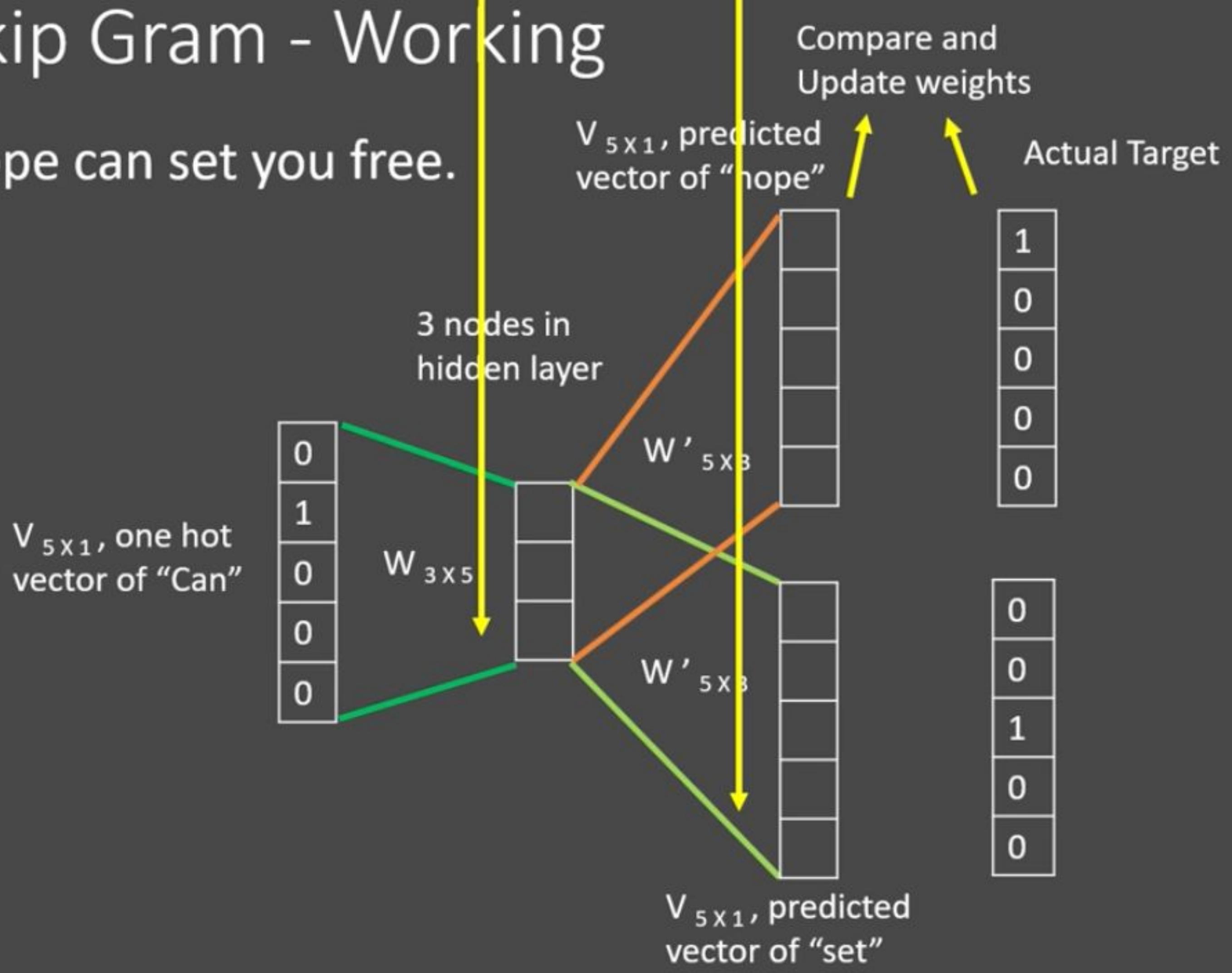
$W'_{5 \times 3}$

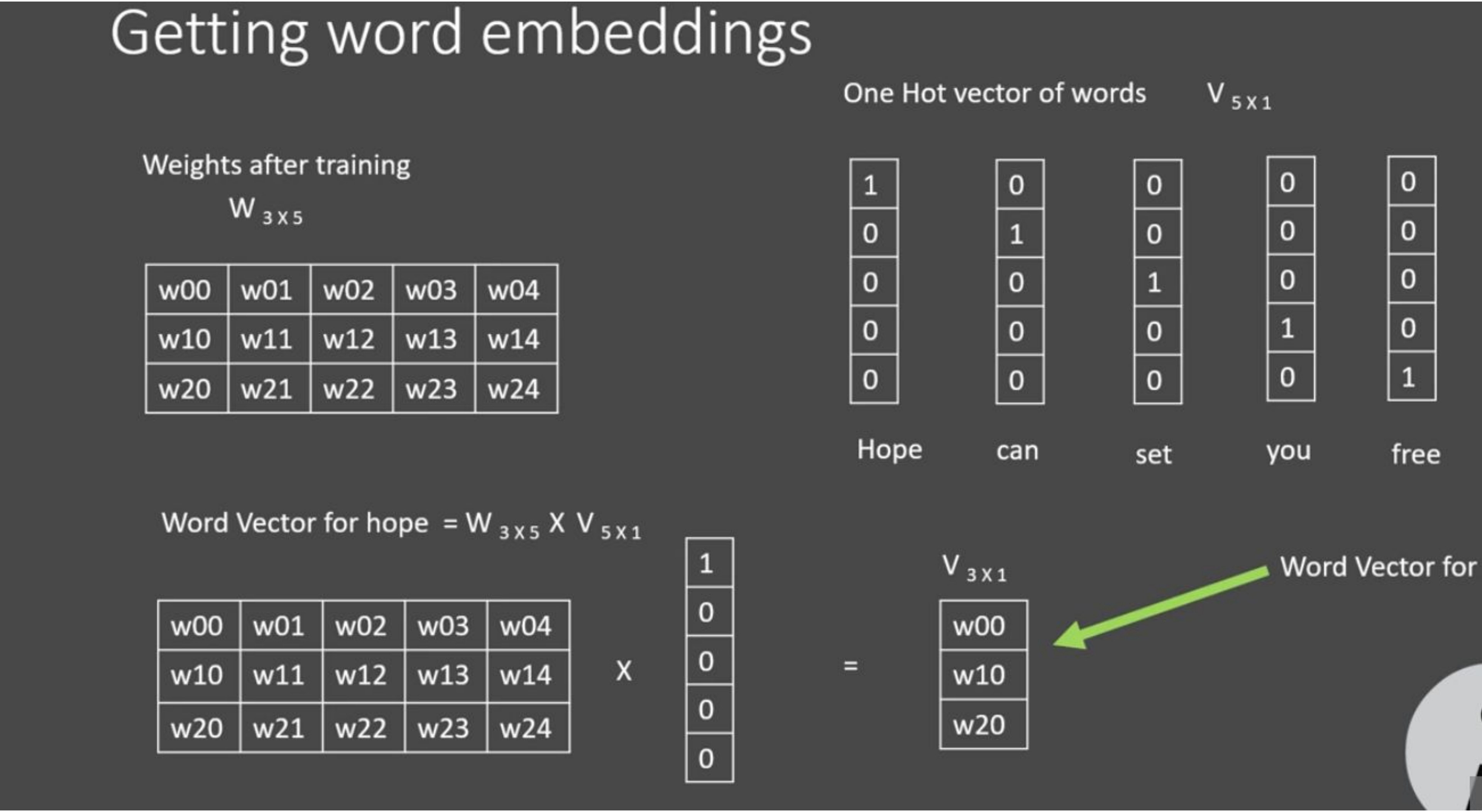
0
0
1
0
0

$V_{5 \times 1}$, predicted vector of “set”

Skip Gram - Working

Hope can set you free.





Getting word embeddings

Weights after training

$W_{3 \times 5}$

$$\begin{bmatrix} w00 & w01 & w02 & w03 & w04 \\ w10 & w11 & w12 & w13 & w14 \\ w20 & w21 & w22 & w23 & w24 \end{bmatrix}$$

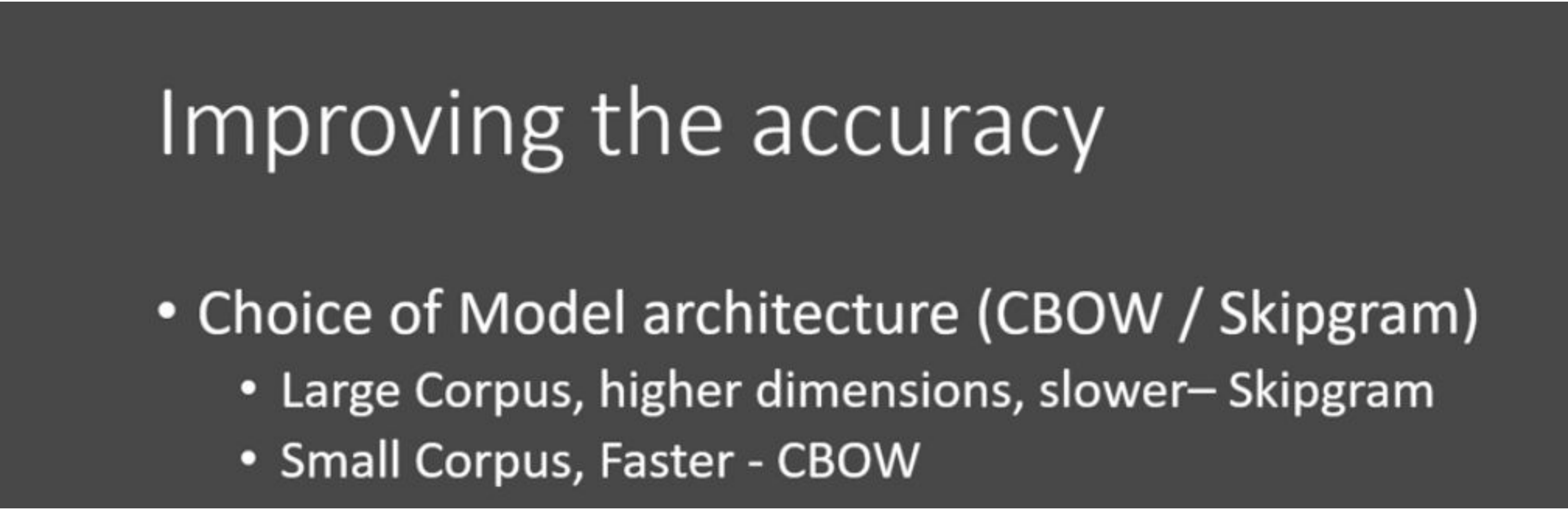
One Hot vector of words $V_{5 \times 1}$

$$\underset{\text{Hope}}{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}} \quad \underset{\text{can}}{\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}} \quad \underset{\text{set}}{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}} \quad \underset{\text{you}}{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}} \quad \underset{\text{free}}{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}$$

Word Vector for hope $= W_{3 \times 5} \text{ X } V_{5 \times 1}$

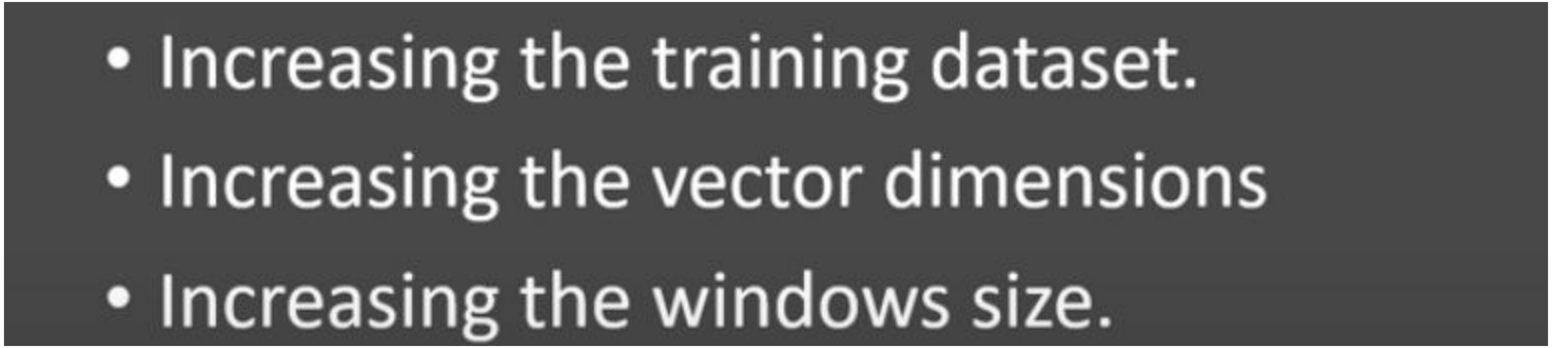
$$\begin{bmatrix} w00 & w01 & w02 & w03 & w04 \\ w10 & w11 & w12 & w13 & w14 \\ w20 & w21 & w22 & w23 & w24 \end{bmatrix} \text{ X } \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \overset{V_{3 \times 1}}{\begin{bmatrix} w00 \\ w10 \\ w20 \end{bmatrix}}$$

Word Vector for



Improving the accuracy

- Choice of Model architecture (CBOW / Skipgram)
  - Large Corpus, higher dimensions, slower– Skipgram
  - Small Corpus, Faster - CBOW



- Increasing the training dataset.
- Increasing the vector dimensions
- Increasing the windows size.