Introduction to Management Information Systems

How Does Artificial Intelligent (AI) work?

Artificial Intelligence

how does AI work?

machine learning

machine learning

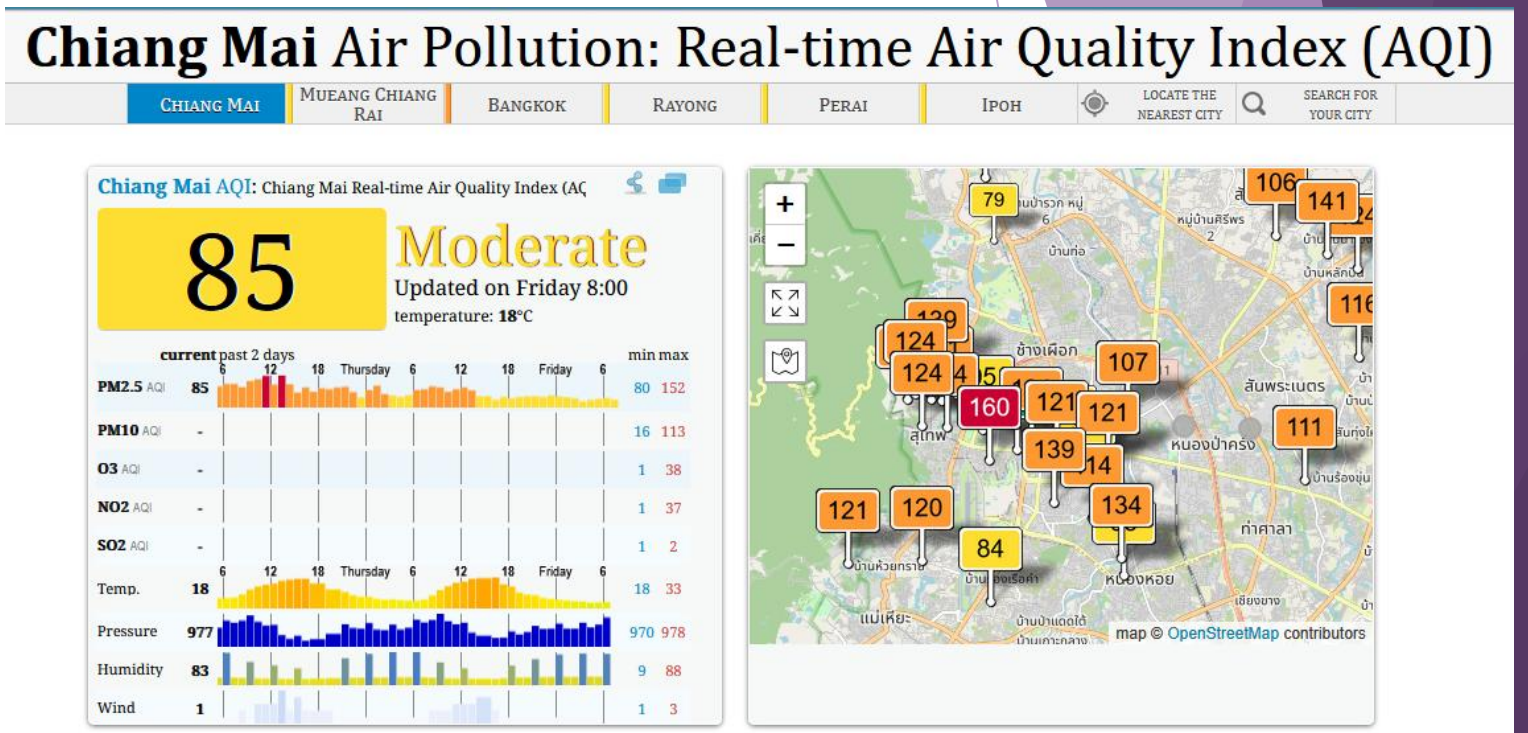
supervised learning

unsupervised learning

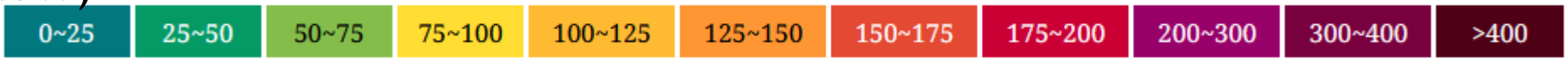
reinforcement learning

regression vs classification

regression
(85)

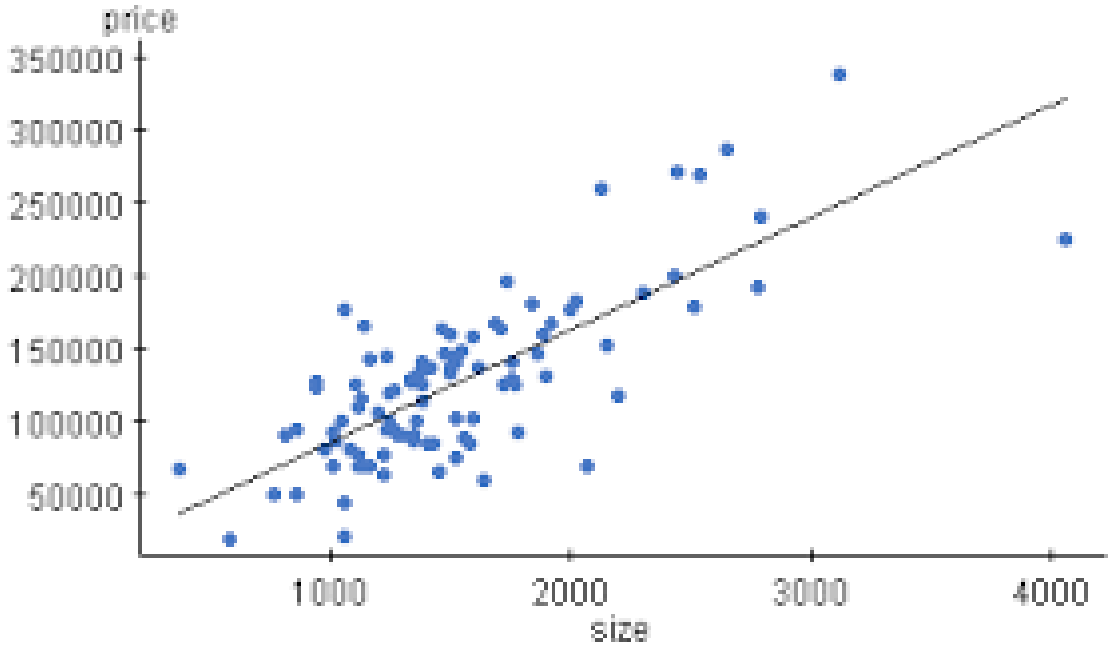


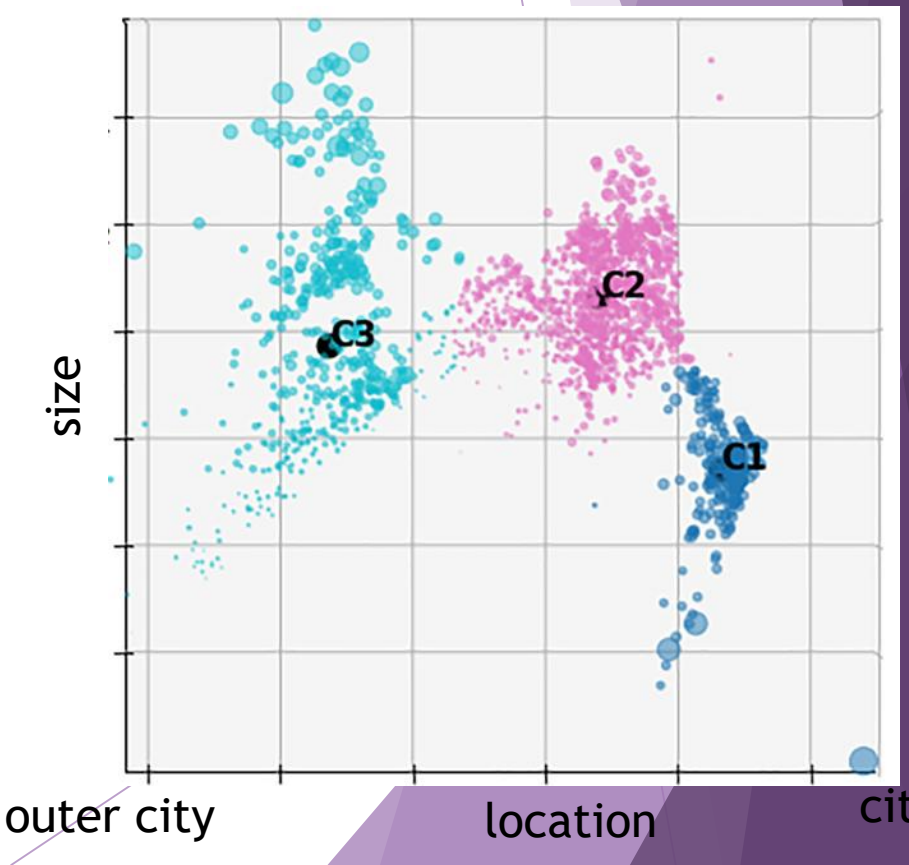
classification
(yellow)



How does machine learning work?

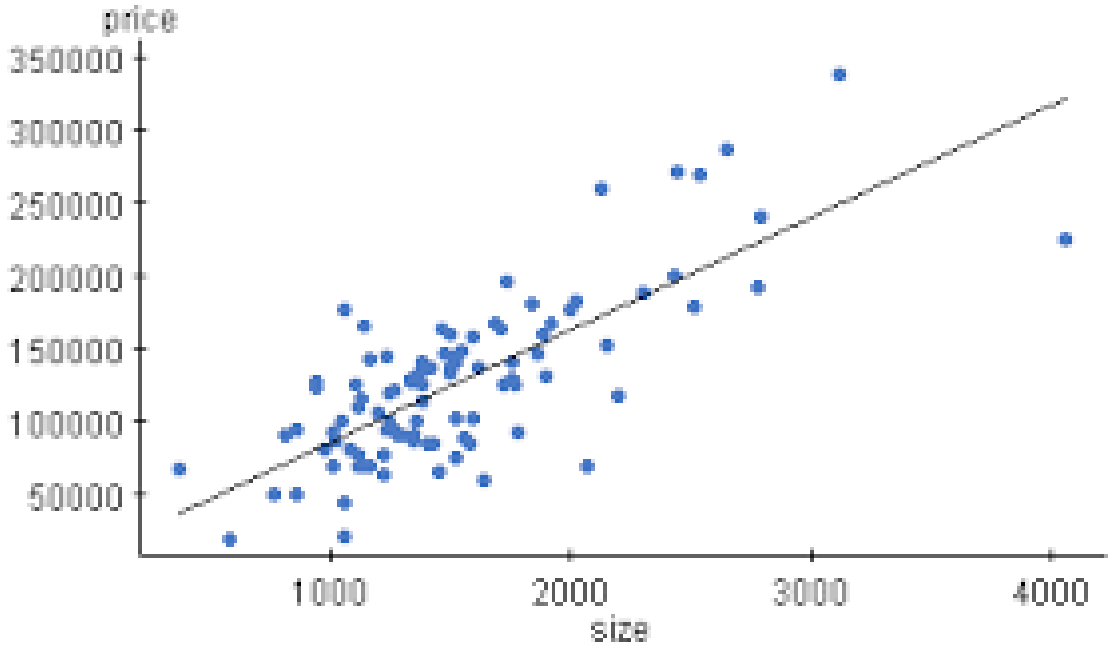
- Think of house prices
- List the factors (features) that decide the price of a house

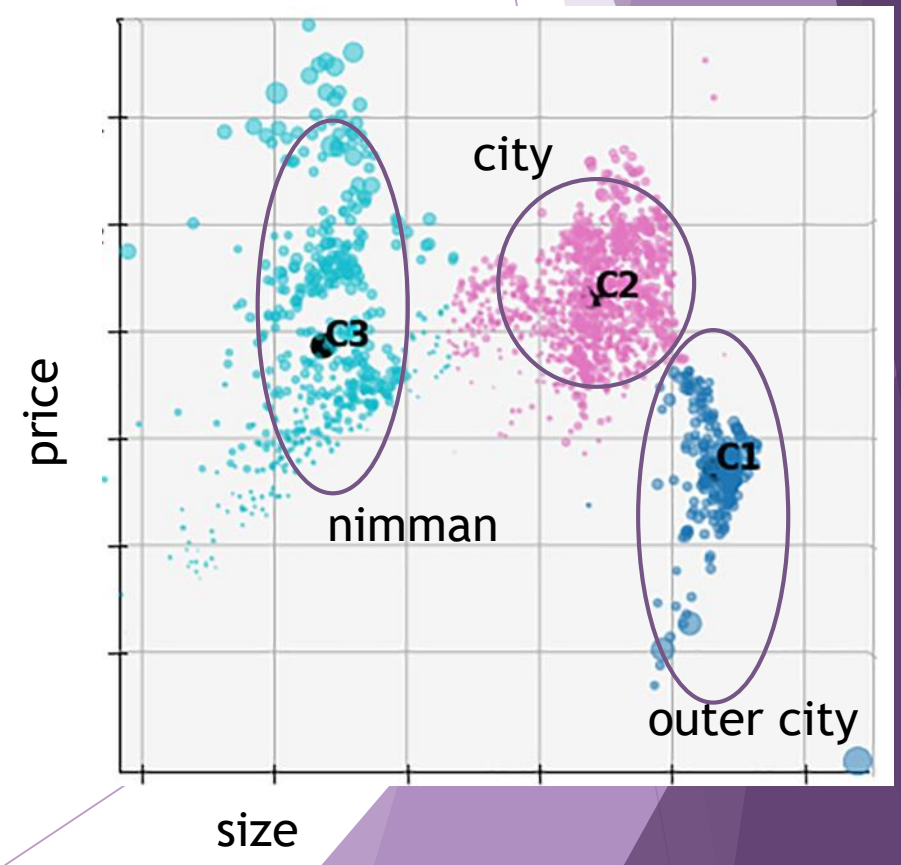




How does machine learning work?

- List the factors (features) that decide the price of a house
- Weights – some factors are more important than others





and text?

text – speech - language

(natural) language (processing) - NLP

- grammar rules
- corpus-based
- statistics / probability from corpus text

then NN exploded, why?

1) more computing power

+

2) internet – more text – training data

before: corpus-based, but not many (freely-available) corpora

also,

3) parallel processing of entire sequences of text

rather than sequentially

(natural) language (processing) - NLP

- grammar rules
- corpus-based
- statistics / probability from corpus text

For example machine translation

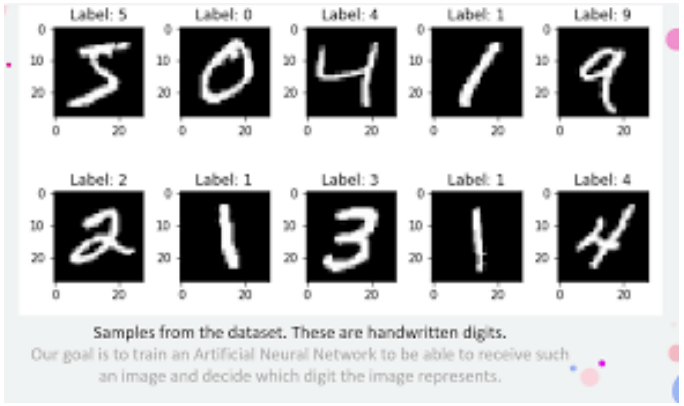
- SMT (statistical MT)
- NMT (neural MT)
  - Google was SMT to GNMT
    - TPUs, internet – training data (3 months)

how does real neural networks work?

https://www.youtube.com/watch?v=HUuUUJktL6E

1.05 to 6.08

digits – determined by few features



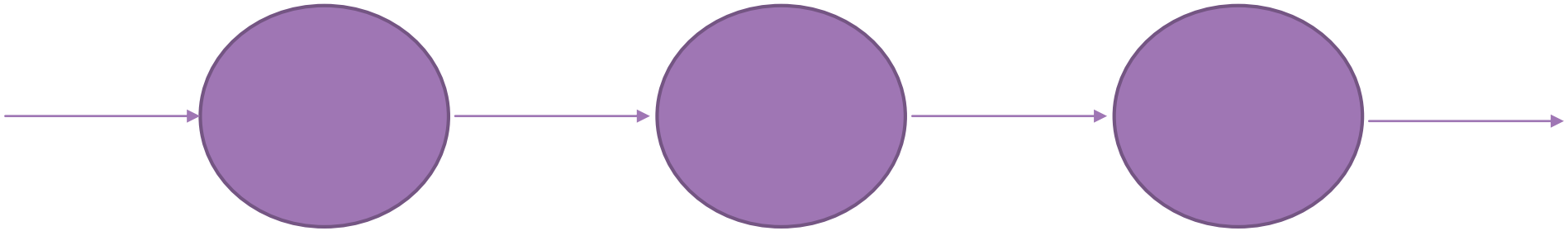
top	line - horizontal	5, 7
	curve	0, 2, 3, 8, 9
	2 straight / angle	4
	line - vertical	1, 6
bottom	curve - horizontal	0, 6, 8
middle	different	0, 8

uses values for pixels e.g. 28*28

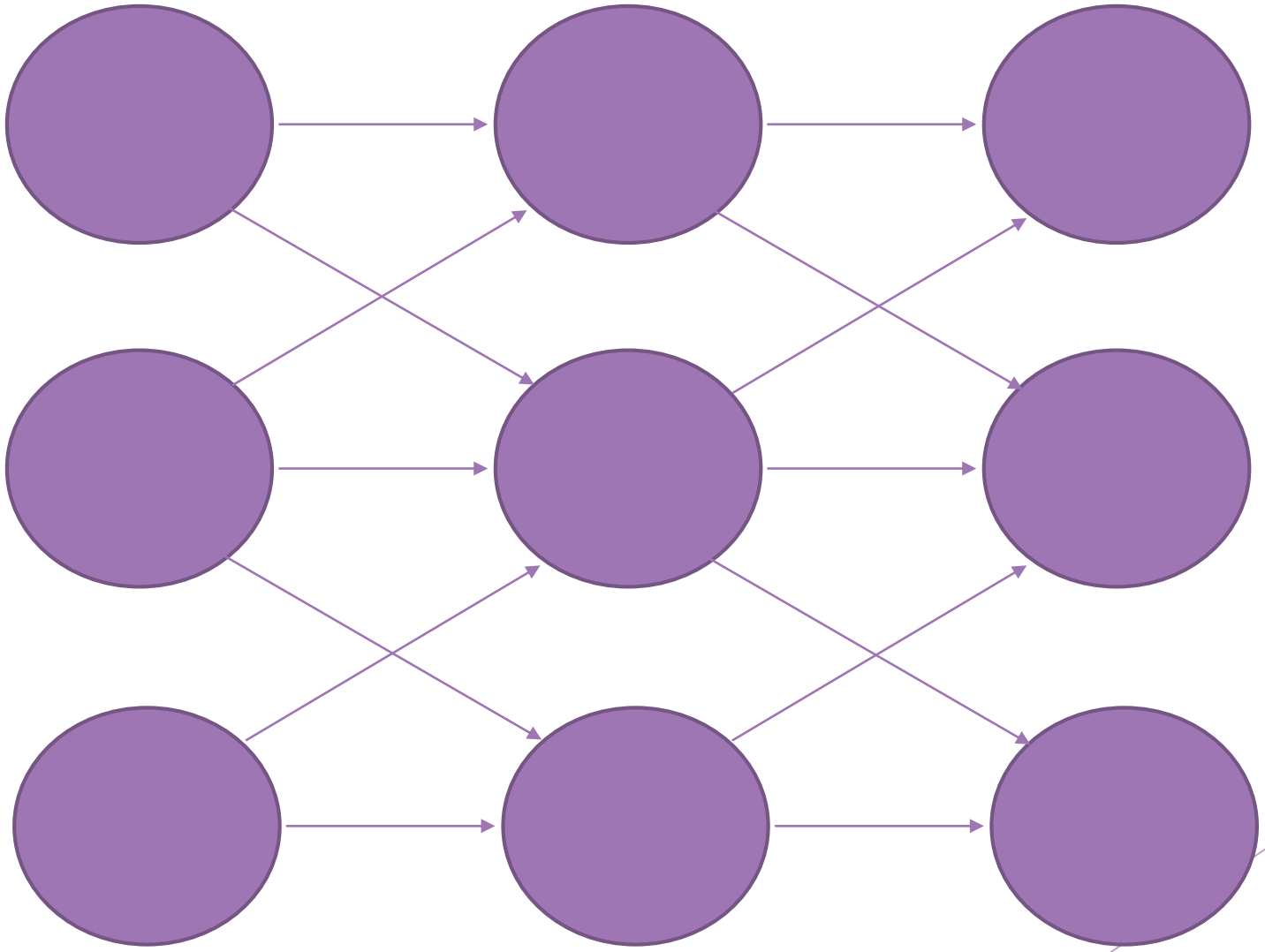
neural networks

Artificial Neural Networks

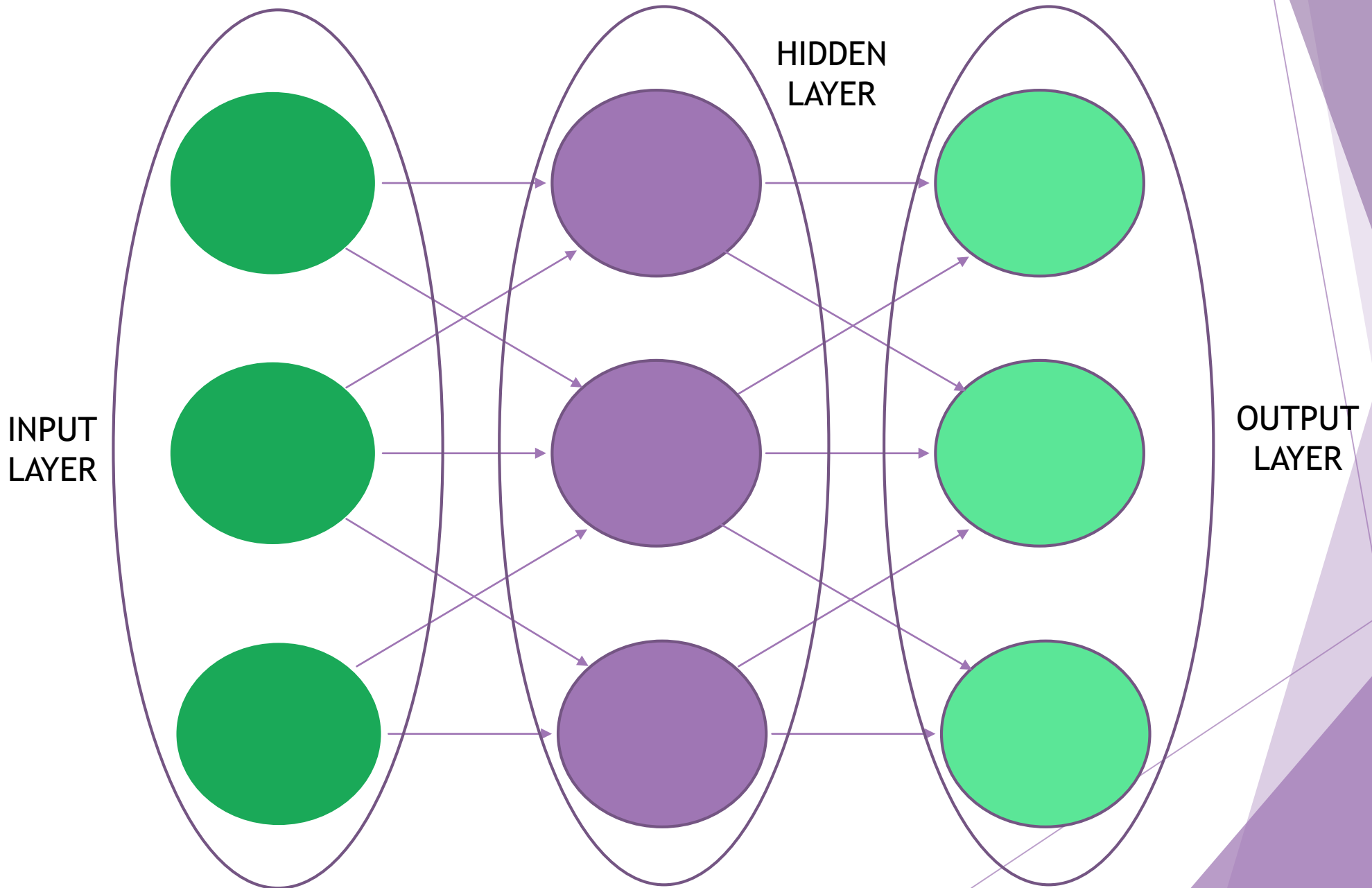
what are neurons?



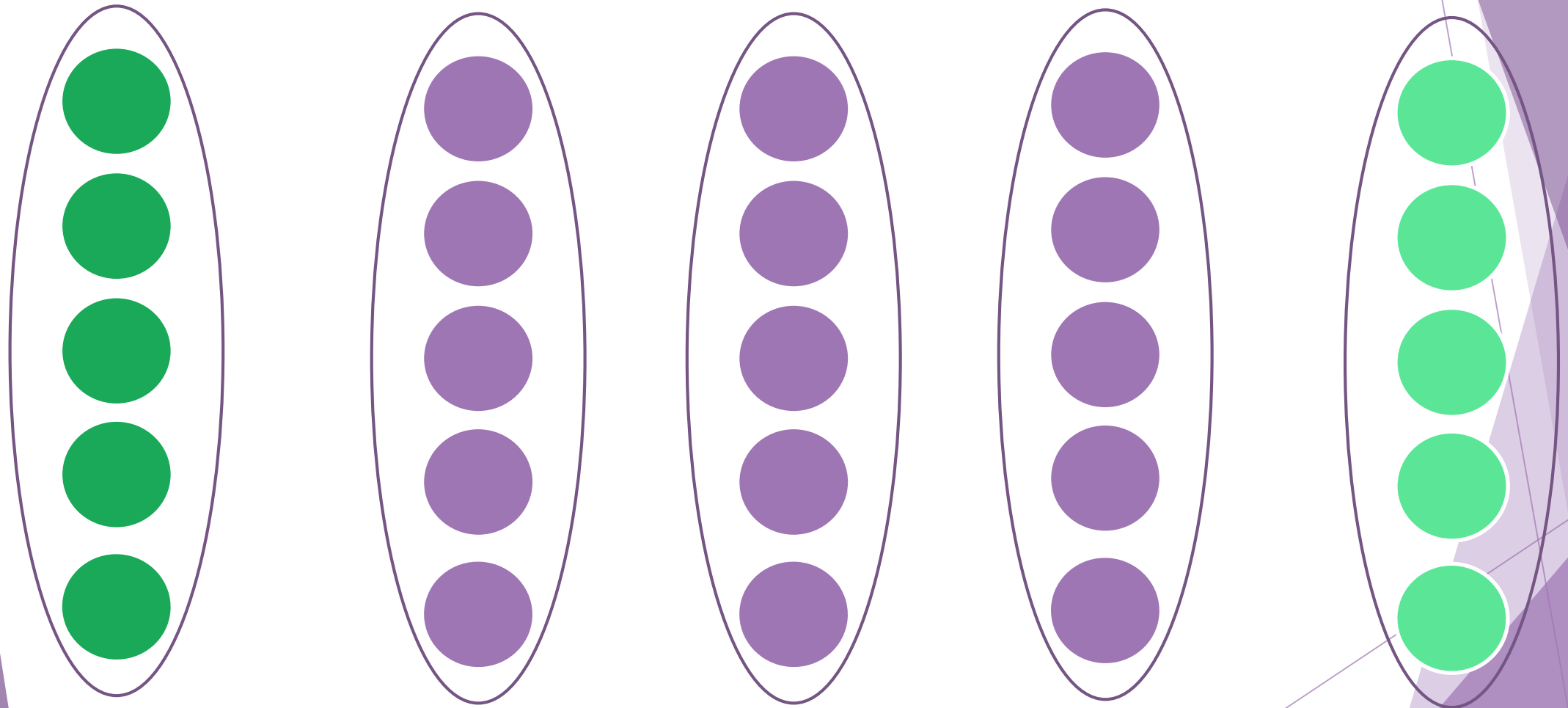
what is a neural network?



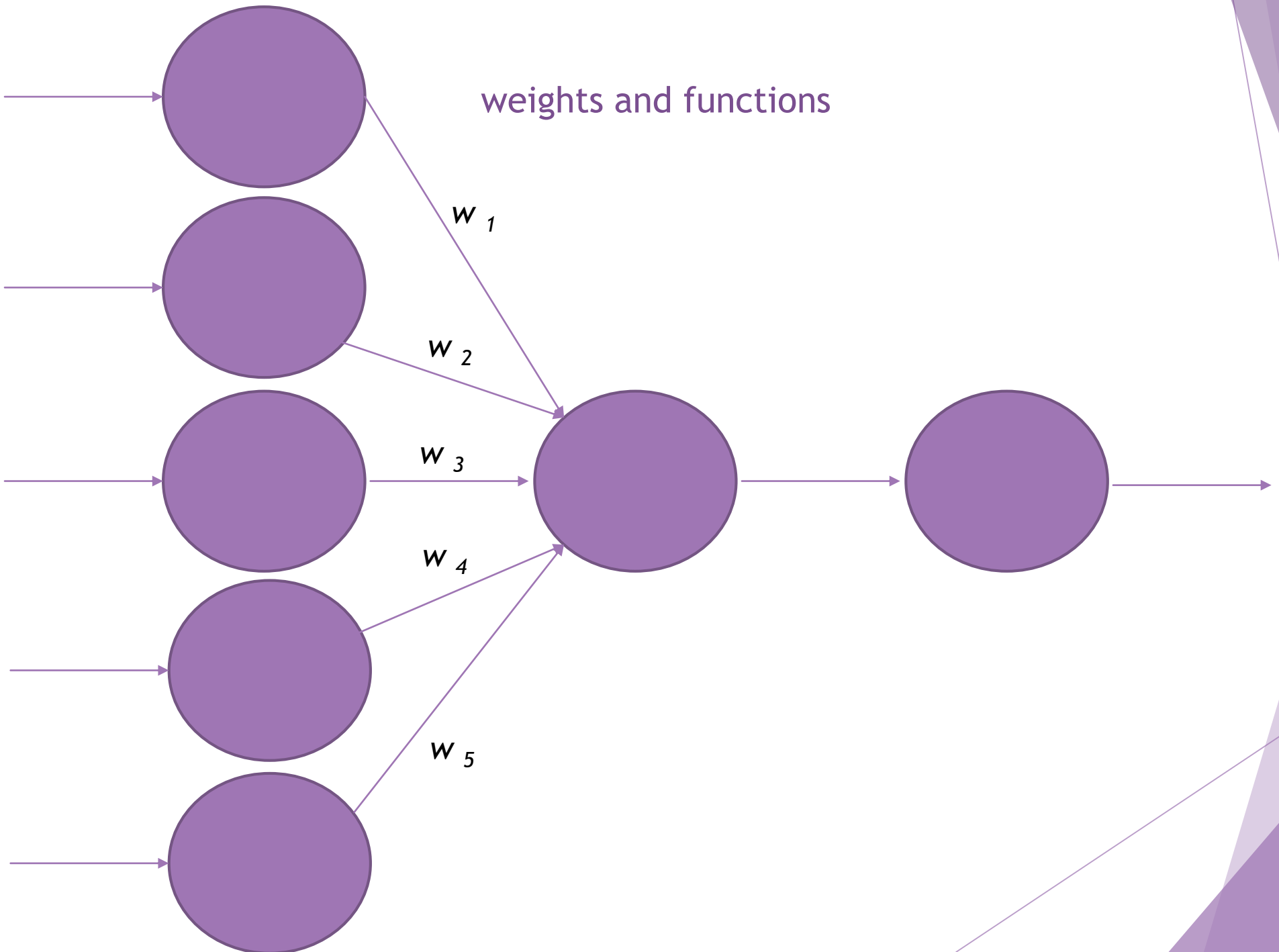
input, output, hidden layer



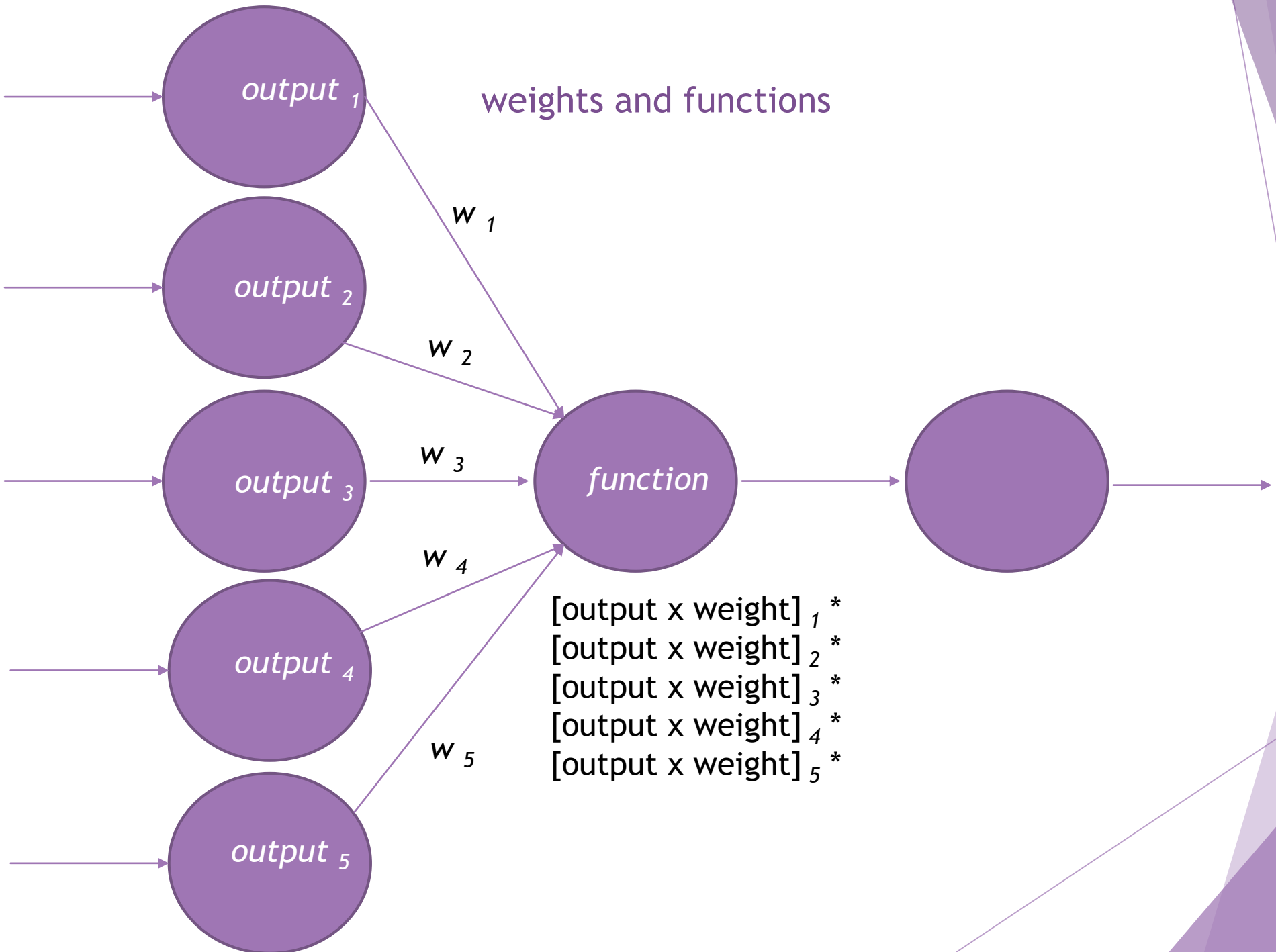
multiple layers = ‘deep’ learning



weights and functions

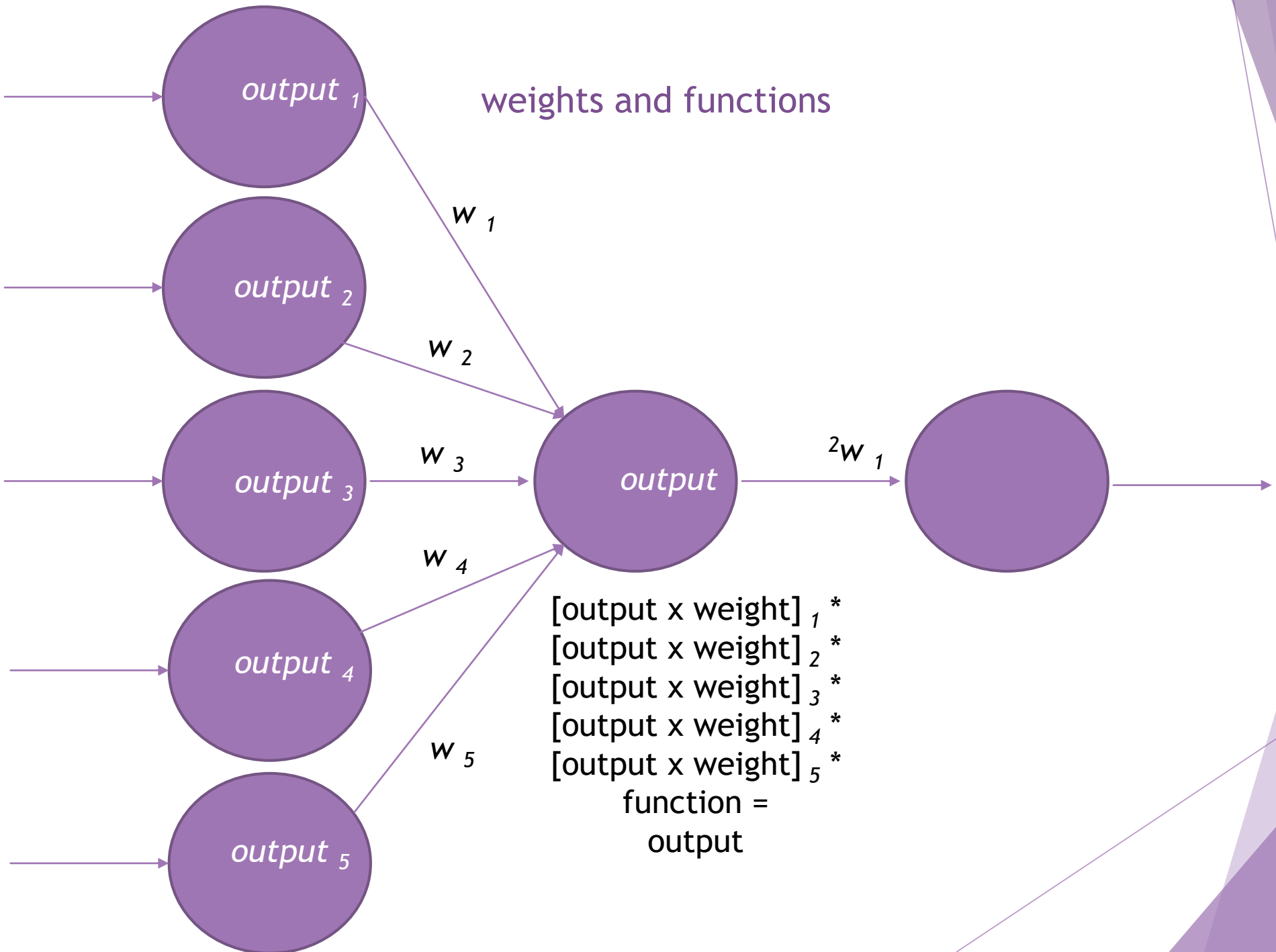


weights and functions



[output x weight] $_1$ *
[output x weight] $_2$ *
[output x weight] $_3$ *
[output x weight] $_4$ *
[output x weight] $_5$ *

weights and functions



how does it work

https://www.youtube.com/watch?v=CqOfi41LfDw

from 8:08 to 16:34

backpropagation

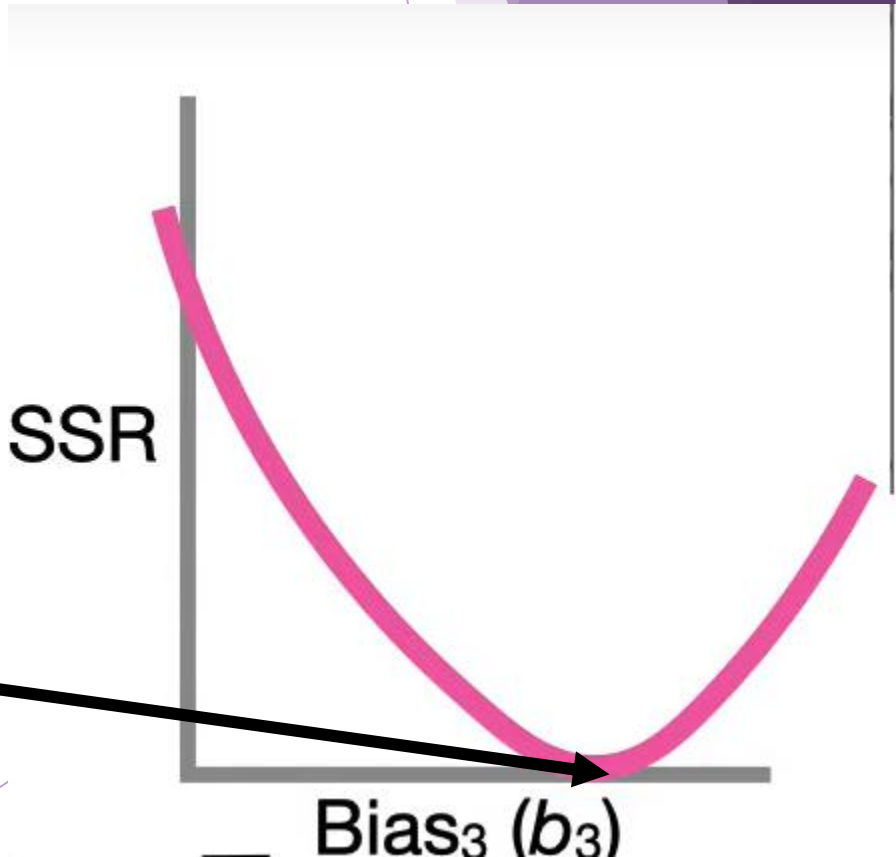
residual = correct value – estimated value

sum of the squared residuals (SSR)

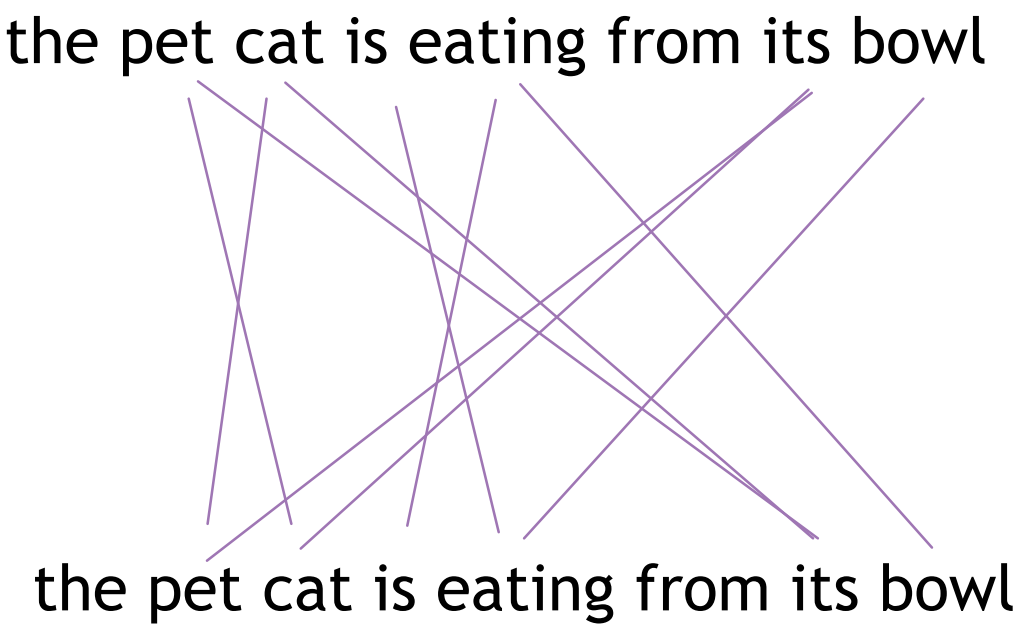
use the chain rule to calculate the derivative of the sum of the squared residuals (SSR)

and gradient descent (function)

to calculate the optimal value

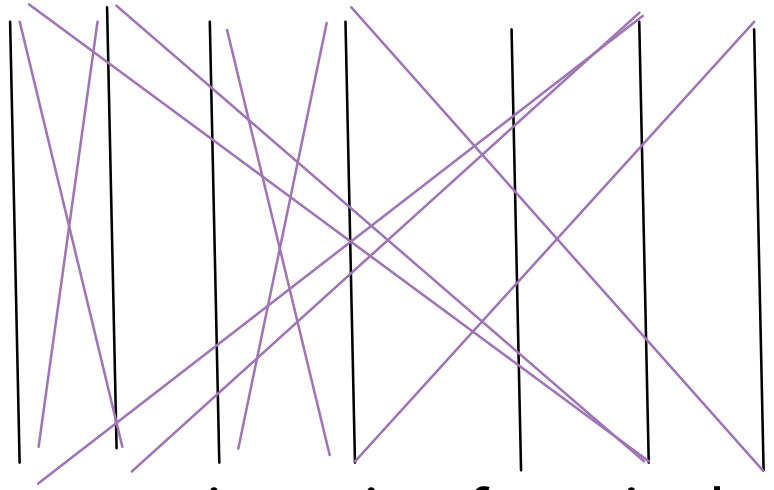


attention is all you need



attention is all you need

the pet cat is eating from its bowl



the pet cat is eating from its bowl

uses query, key and
value vectors to focus
on term similarity

language

but language or text is a sequence of words

the sequence is important

the order is important

“the man ran down the car”

is not the same as

“the car ran down the man”

transformers

sentence order

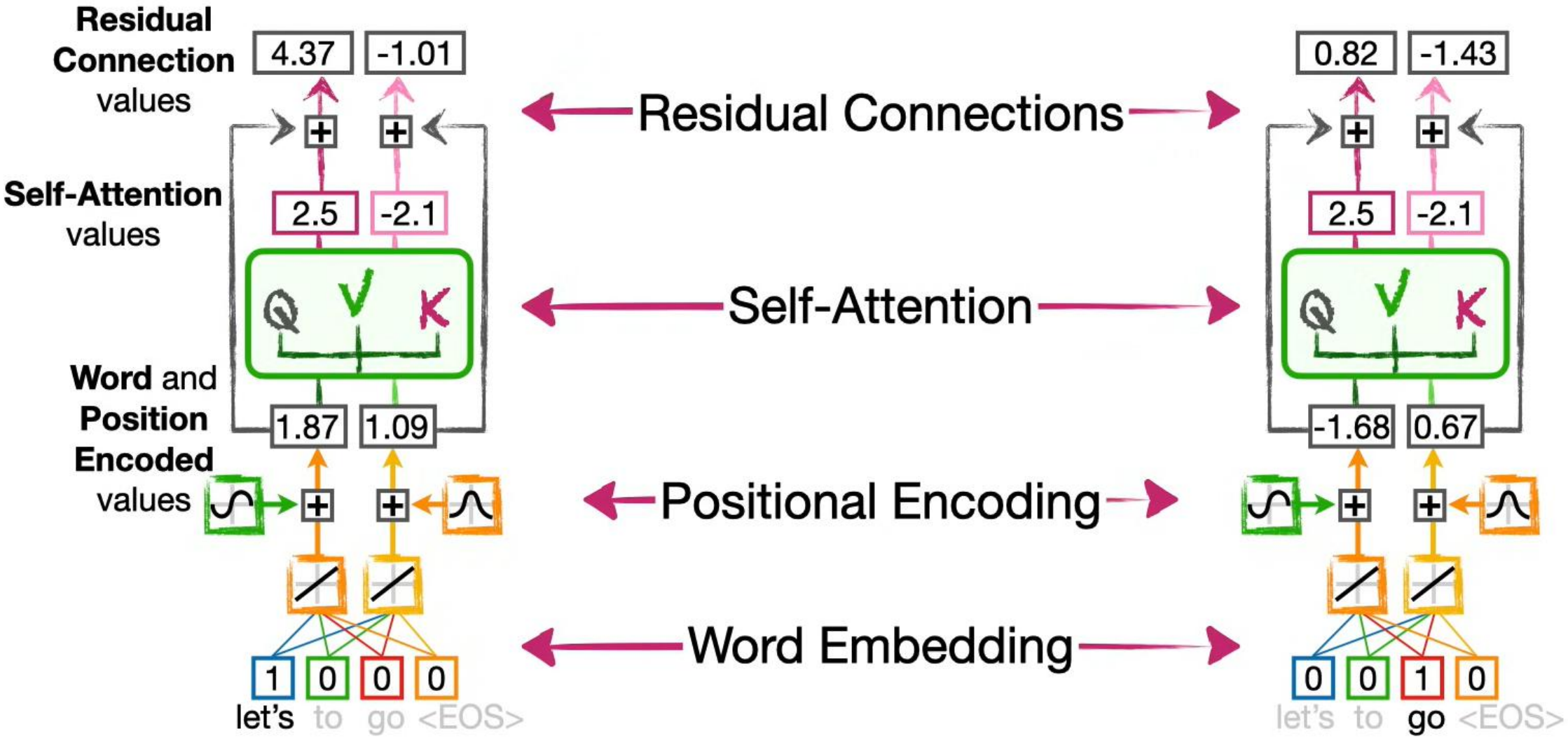
- positional encoding – weights for position to reflect order

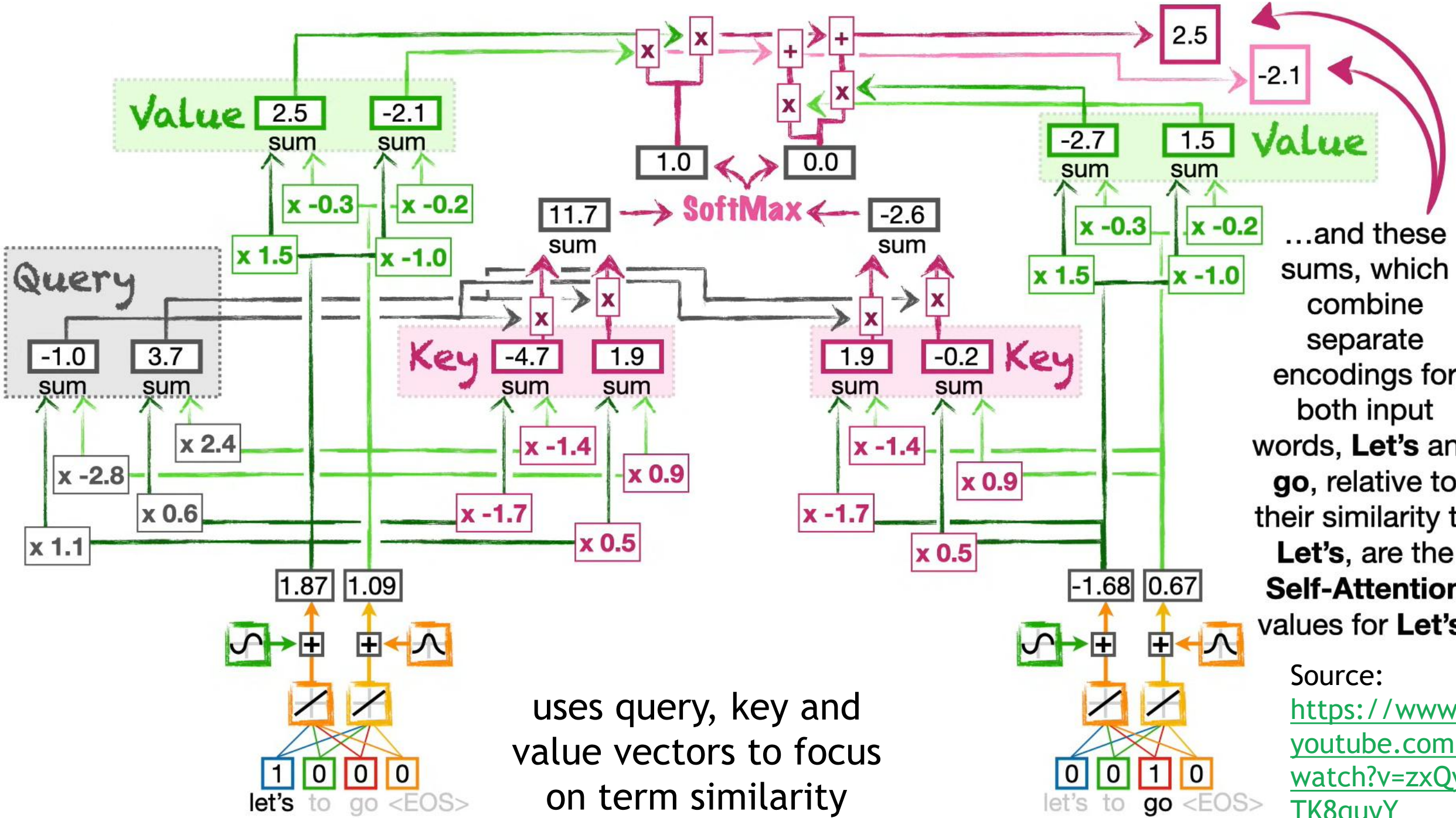
relationship between words

- self-attention – learn relationship between words by how they are used in the content of the training data

uses a SoftMax function for the relative similarity

NOTE: This simple **Transformer** only contains the parts required for encoding the input.




Value
2.5
sum
-2.1
sum
x -0.3
x -0.2
x 1.5
x -1.0
Query
-1.0
sum
3.7
sum
x 2.4
x -2.8
x 0.6
x 1.1
1.87
1.09
1
0
0
0
let's
to
go
<EOS>
1.0
0.0
11.7
sum
SoftMax
-2.6
sum
Key
-4.7
sum
1.9
sum
x -1.4
x 0.9
x -1.7
x 0.5
1.9
sum
-0.2
sum
Key
x -1.4
x 0.9
x -1.7
x 0.5
2.5
-2.1
-2.7
sum
1.5
sum
Value
x -0.3
x -0.2
x 1.5
x -1.0
-1.68
0.67
0
0
1
0
let's
to
go
<EOS>
…and these sums, which combine separate encodings for both input words, Let's and go, relative to their similarity to Let's, are the Self-Attention values for Let's.
uses query, key and value vectors to focus on term similarity
Source: https://www.youtube.com/watch?v=zxQyTK8quyY

encoders - decoders

A review of Thai–English machine translation

Séamus Lyons[1]

Received: 11 January 2019 / Accepted: 20 July 2020 / Published online: 14 August 2020

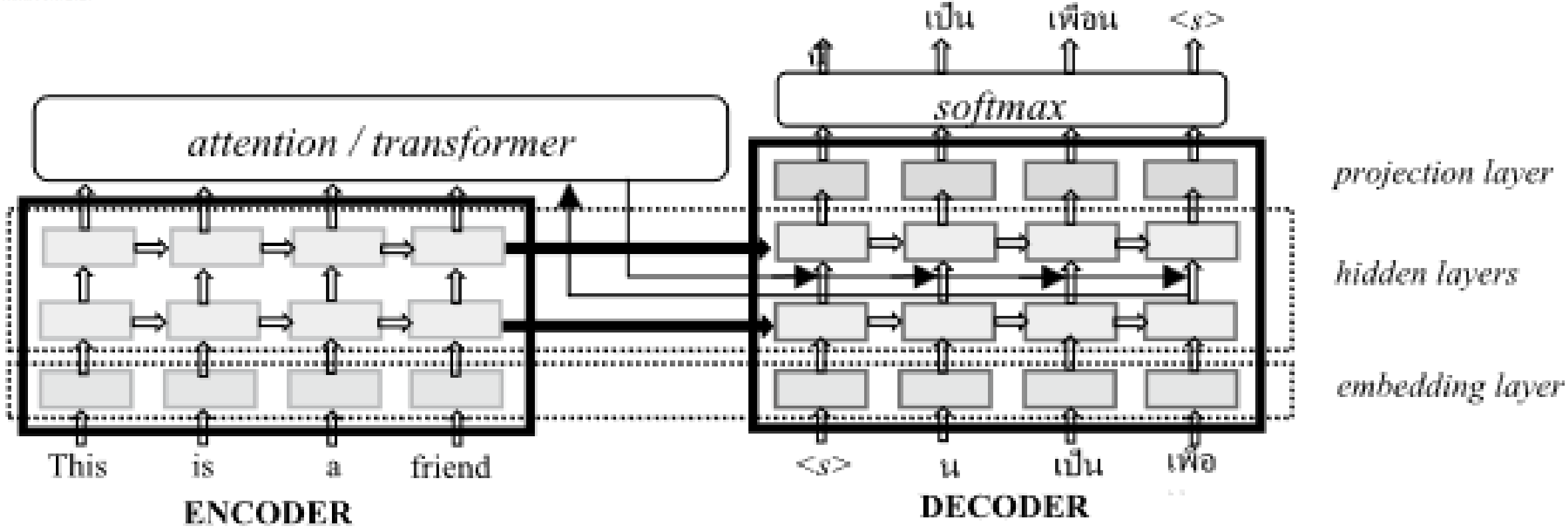


Fig. 5 A neural machine translation (NMT) system

Transformers

Model - create output

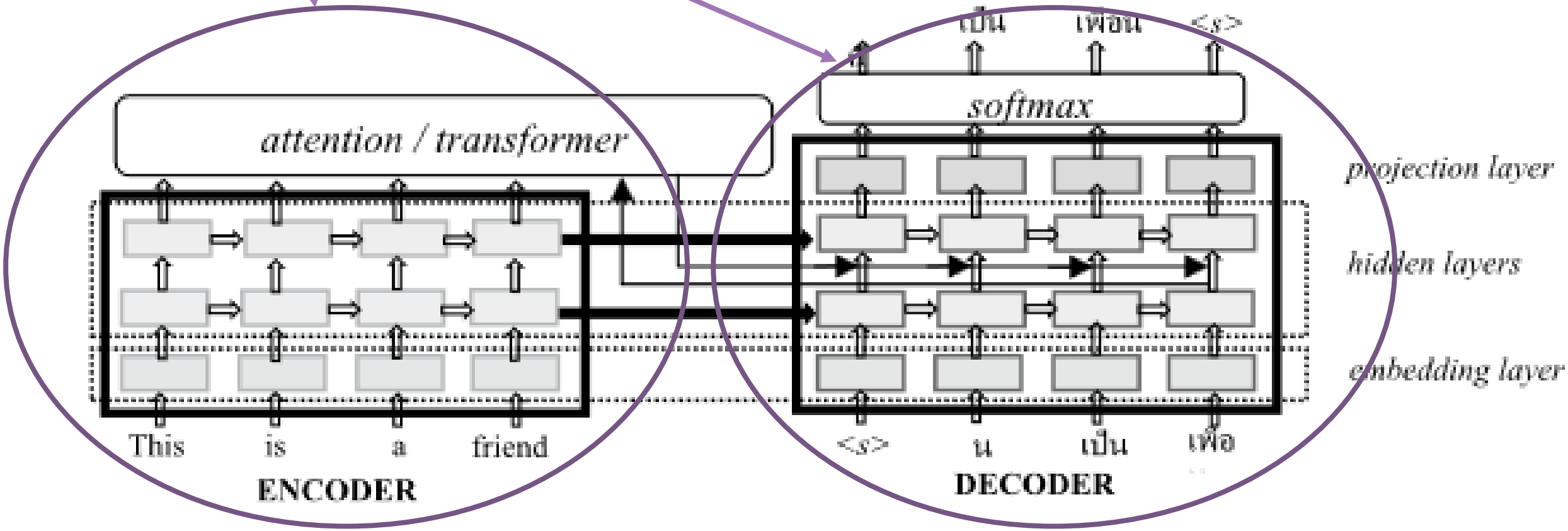
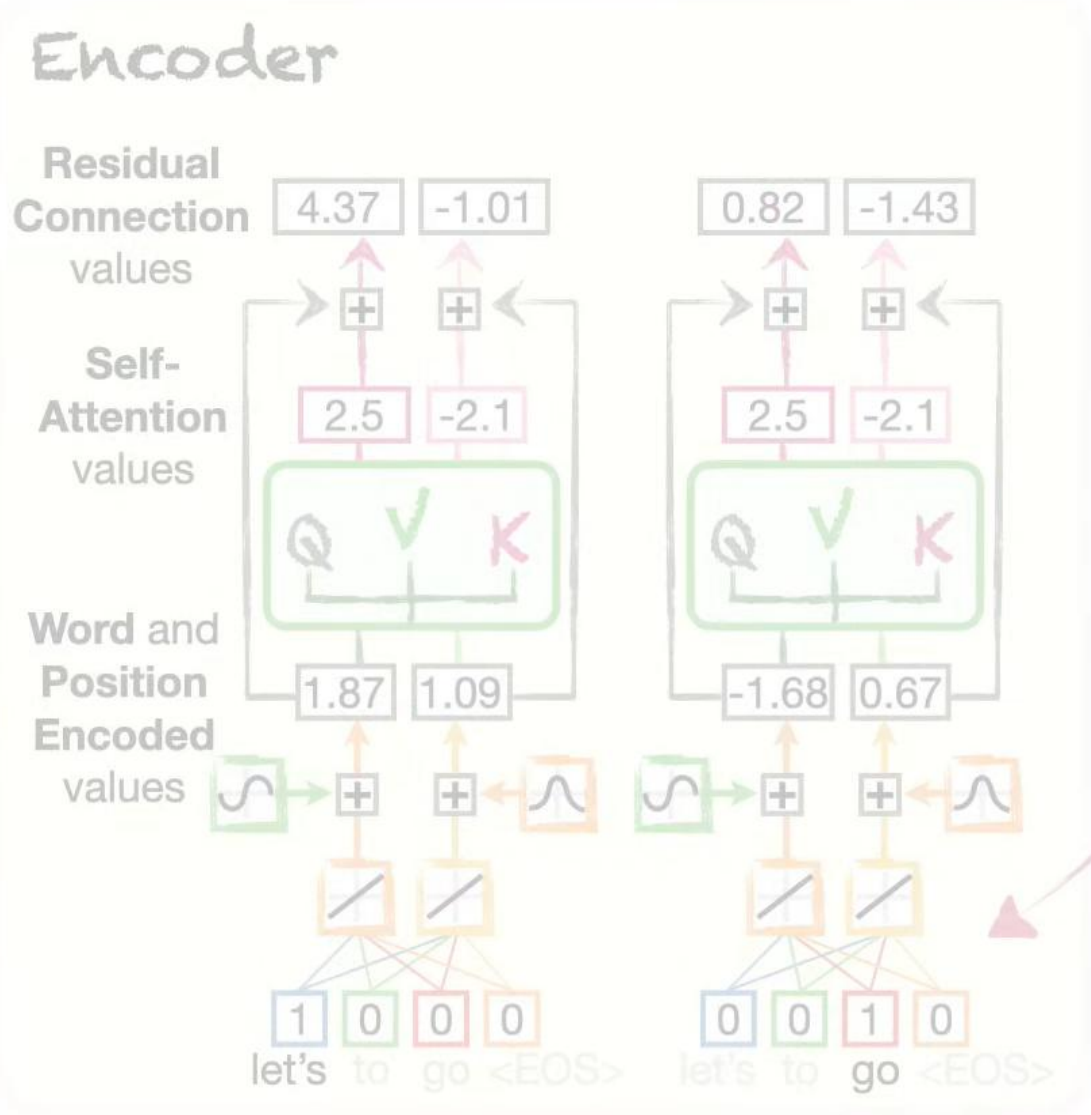
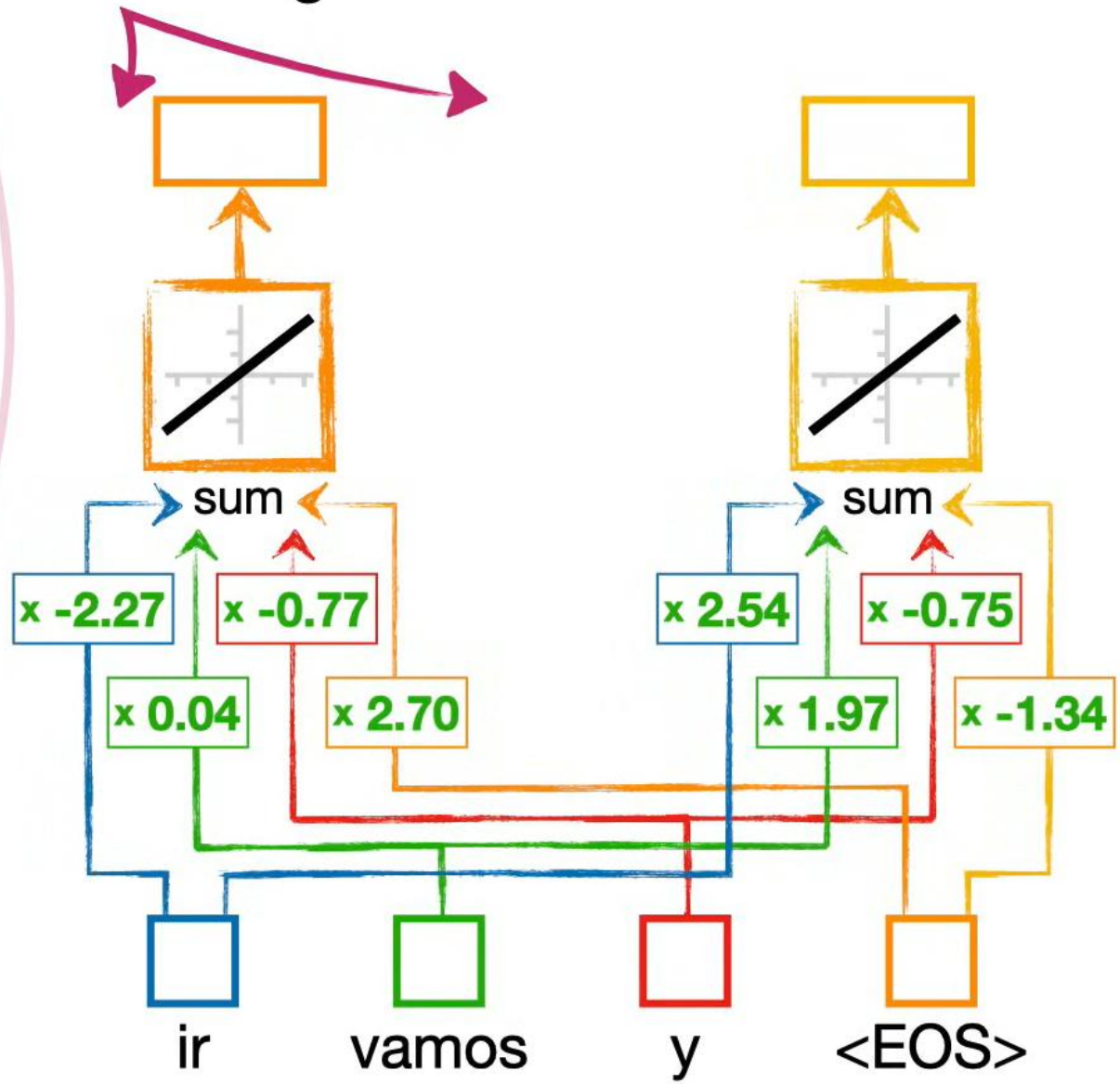


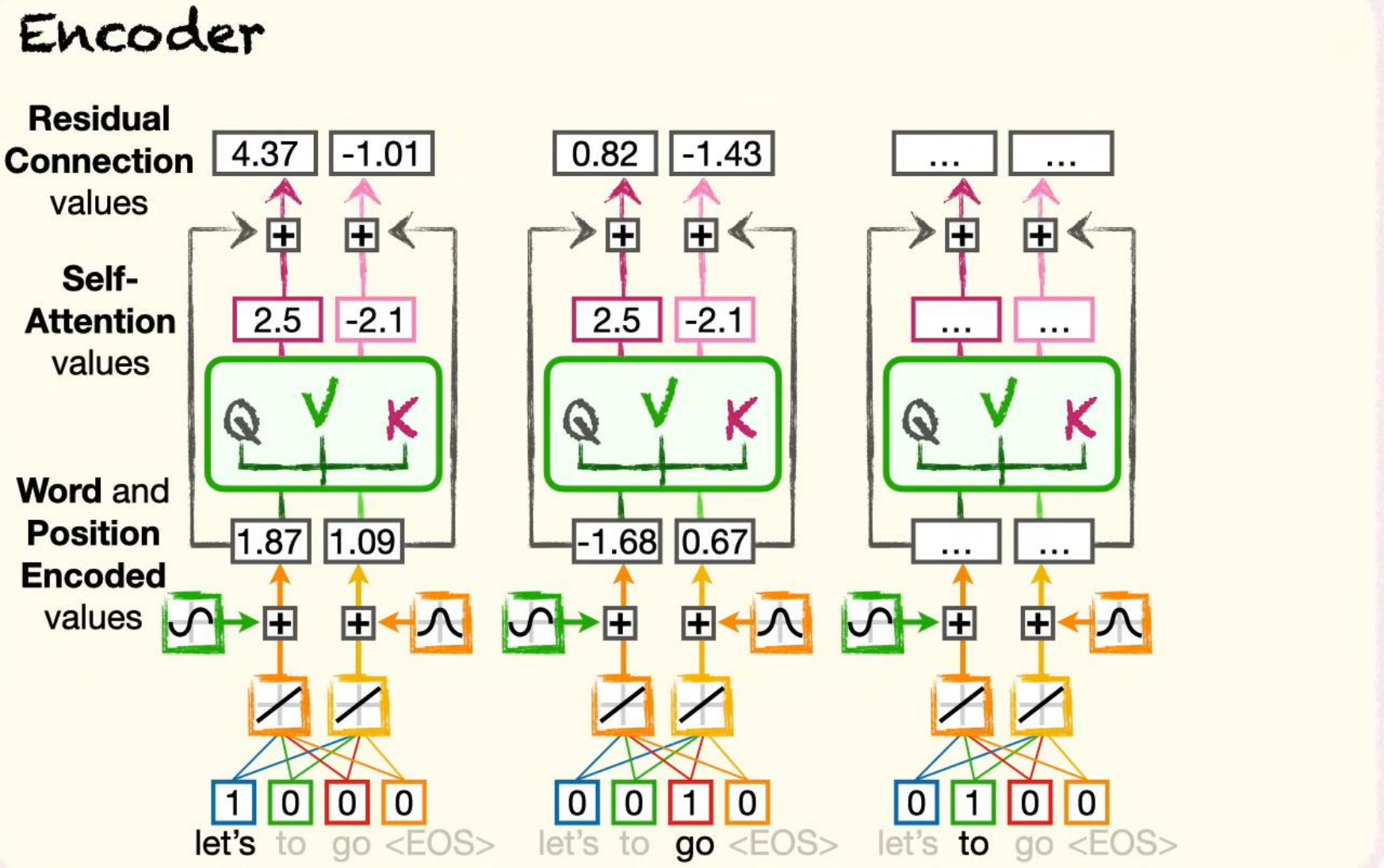
Fig. 5 A neural machine translation (NMT) system

The **Decoder**, just like the **Encoder**, starts with **Word Embedding**.





Doing all of the computations at the same time, rather than doing them sequentially for each word…



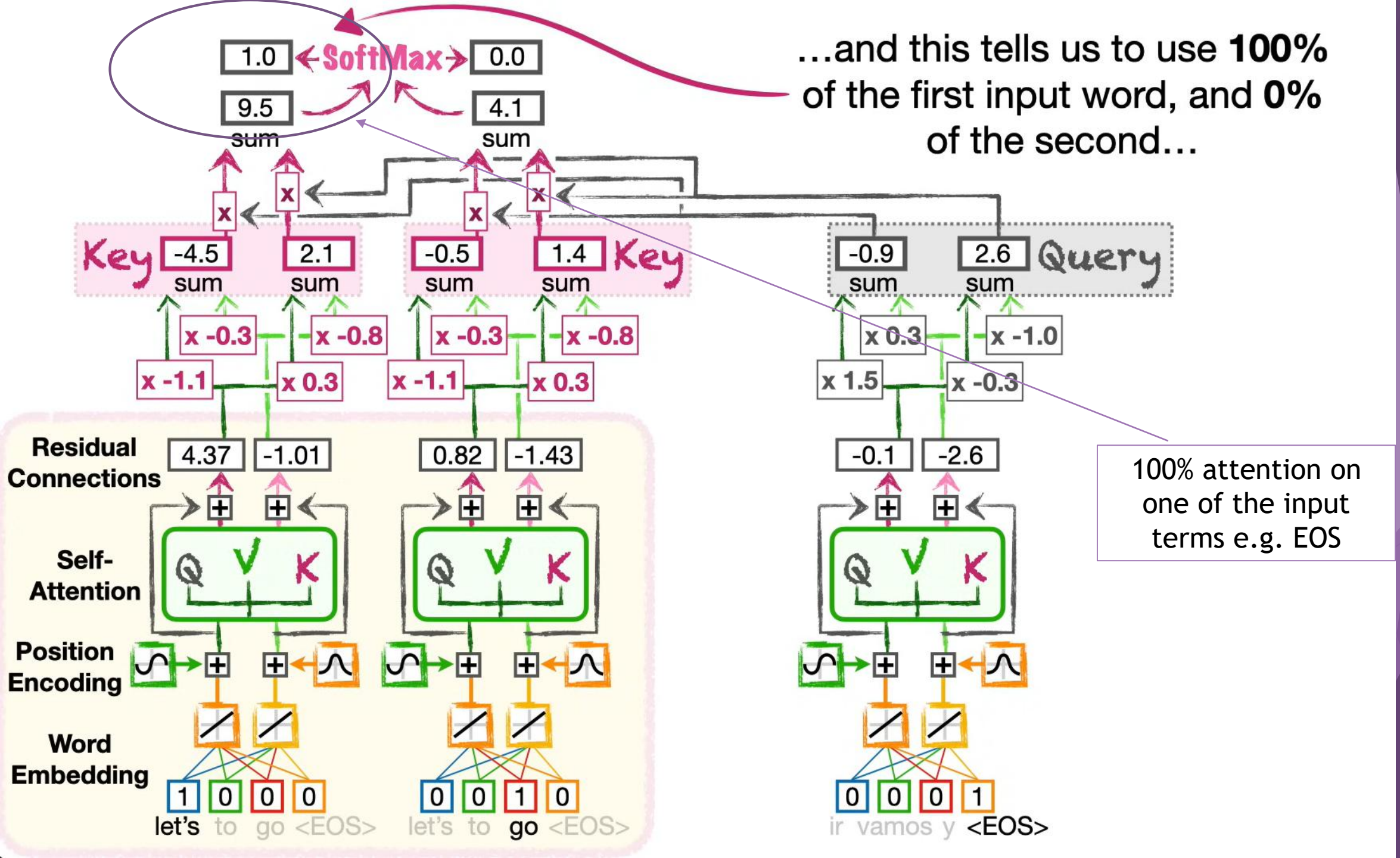
…means we can process a lot of words relatively quickly on a chip with a lot of computing cores, like a **GPU** (graphics processing unit), or on multiple chips in the cloud.

So, the main idea of **Encoder-Decoder Attention** is to allow the **Decoder** to keep track of the significant words in the input.

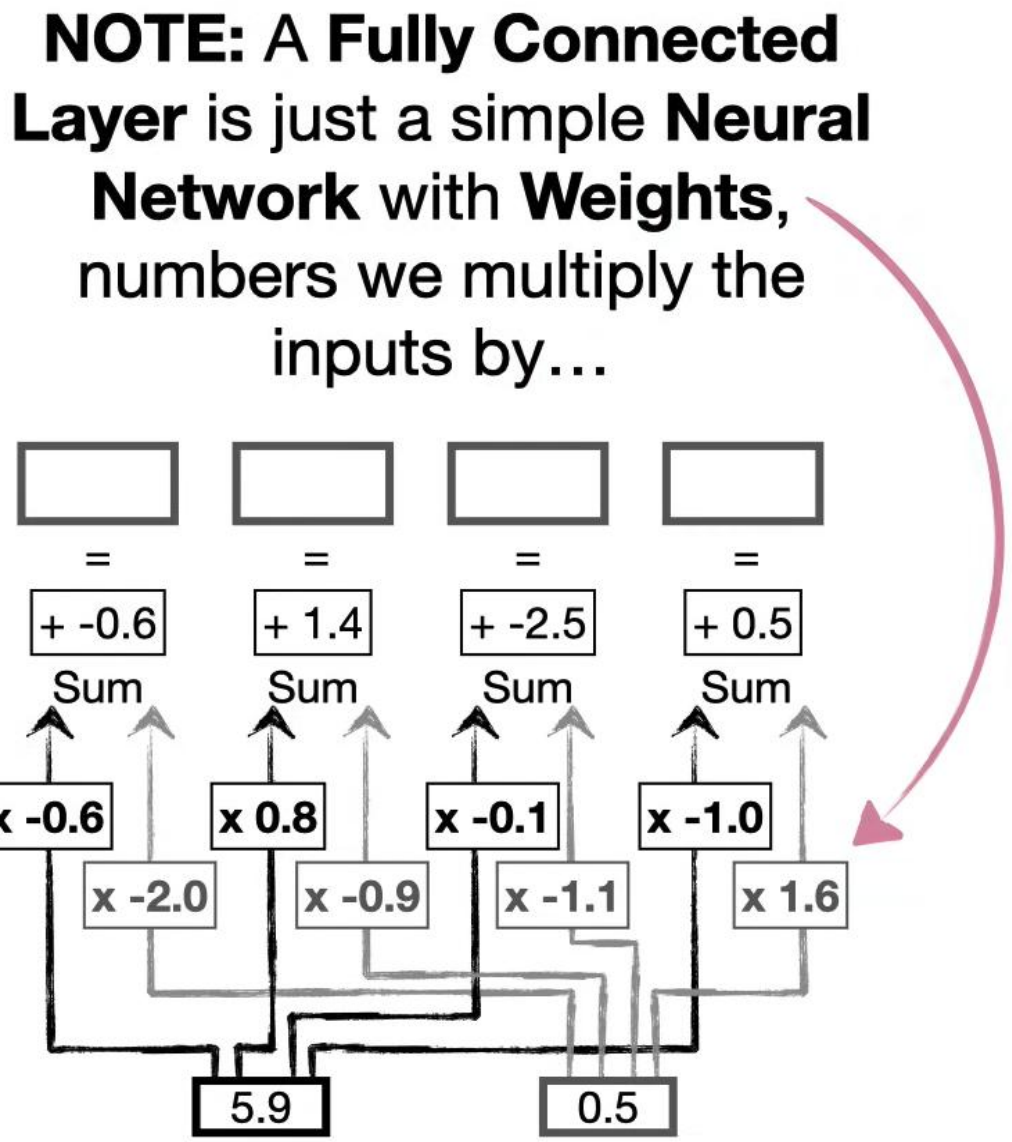
need to focus on the term don’t

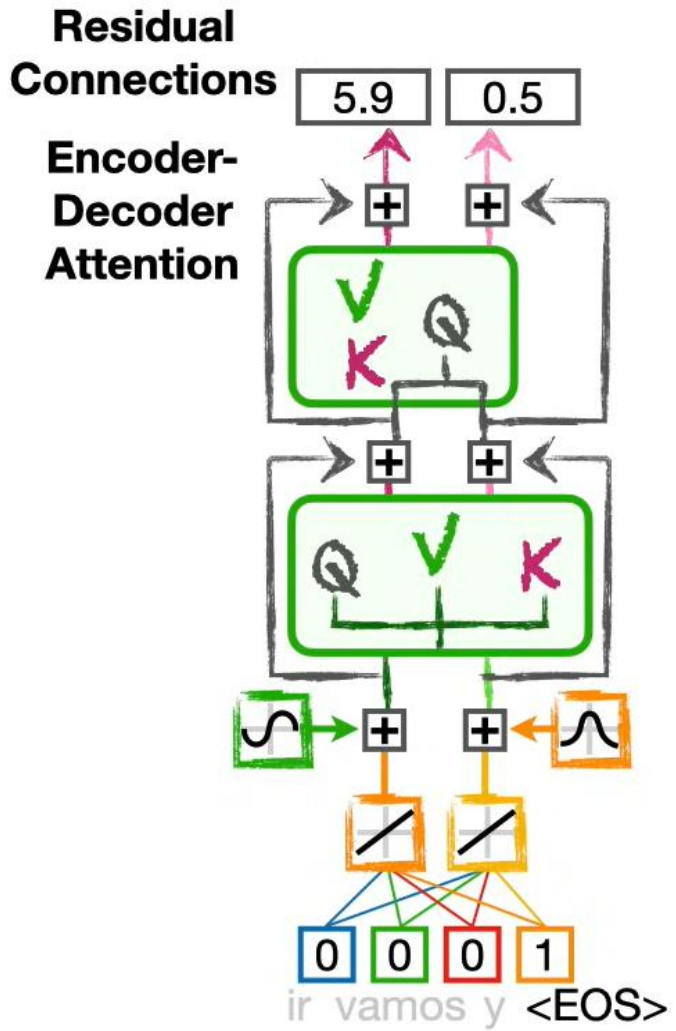
Don’t eat the delicious looking and smelling pizza.

Eat the delicious looking and smelling pizza.


1.0
SoftMax
0.0
9.5
sum
4.1
sum
…and this tells us to use 100% of the first input word, and 0% of the second…
Key
-4.5
sum
2.1
sum
-0.5
sum
1.4
sum
Key
-0.9
sum
2.6
sum
Query
x -0.3
x -0.8
x -0.3
x -0.8
x 0.3
x -1.0
x -1.1
x 0.3
x -1.1
x 0.3
x 1.5
x -0.3
Residual Connections
4.37
-1.01
0.82
-1.43
-0.1
-2.6
Self-Attention
Q
V
K
Position Encoding
Word Embedding
1 0 0 0
let's to go <EOS>
0 0 1 0
let's to go
0 0 0 1
ir vamos y <EOS>
100% attention on one of the input terms e.g. EOS

output – another ANN





summary

https://www.youtube.com/watch?v=zxQyTK8quyY

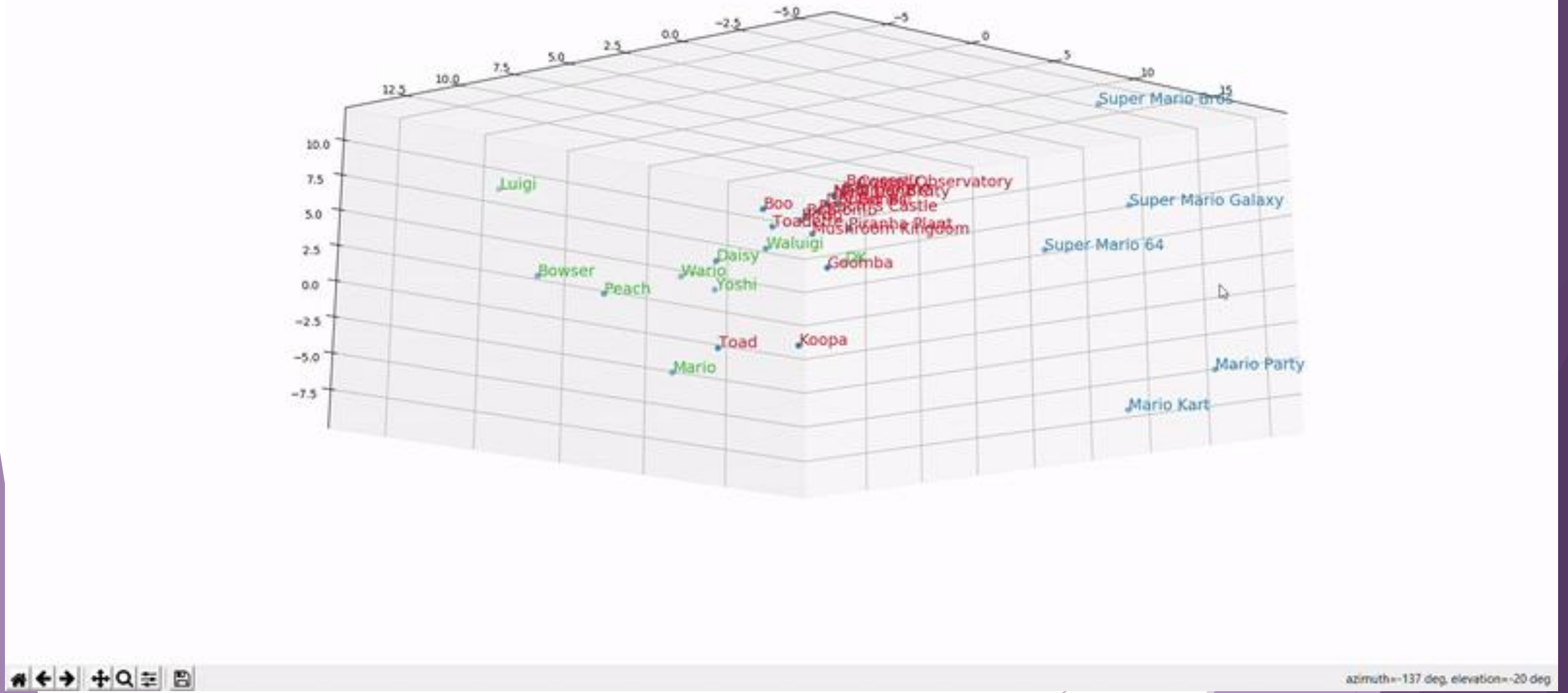
to 33:40 to 34.24

large language models

part 1 pre training - the transformer

input

- text is a sequence of tokens (e.g. words, digits, characters etc.)
- but a neural network needs numbers
- how about 1,2,3 ?
- but is word 2 next to 1 and before word 3?
- one-hot encoding
- now more advanced
  - encoding
  - word2vec



Source: https://medium.com/@marcusa314/visualizing-words-377624cb20c7

backpropagation

- nodes / neurons are connected by edges
- learning is done by the use of numbers
  - input * weight + bias
- these are set randomly
- but are changed to fit the data by backpropagation

backpropagation

training process

- loop - keep using the training data
  - calculations are made using random values
  - then compared to the actual values
  - residual = actual value – estimated value
  - the weights and biases are altered

until all the residuals are low & do not change

so the weights and biases represent the input data

architecture (decoder-only chatGPT)

words to numbers

(word encoding e.g. weights – backpropagation)

position

(positional encoding)

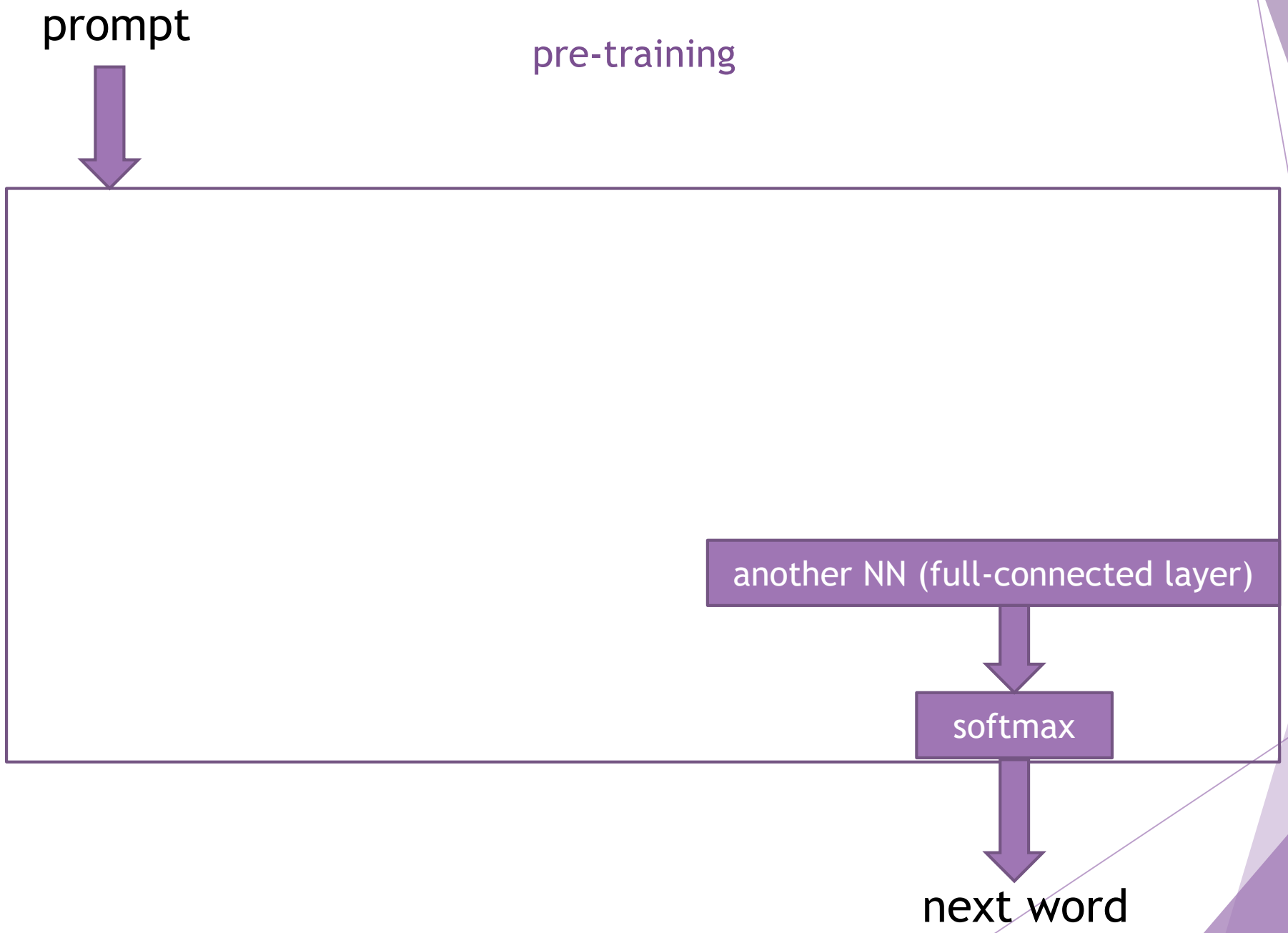
attention

(self-attention – similarity to sentence terms e.g. it)

residuals

addition that helps performance

diagrams (https://www.youtube.com/watch?v=bQ5BoolX9Ag)


prompt
pre-training
another NN (full-connected layer)
softmax
next word

part 2
answering the prompt

generative

- Language models learn texts meaning
- traditionally to predict the next word in a sequence
- LLMs do this in pretraining

then

- special datasets are used for more training
- such a question-answering facility like a Q & A system
  - Standard Question Answering Dataset (SQuAD) – 100K+
  - Google/Bing/other Q&A Datasets
  - Natural questions – from web
  - Multi-Lingual datasets

Q & A training

Large Language Models (LLMs) can understand questions that are similar to prompts and use examples to generate responses.

This capability stems from their training and architecture

Q & A training

LLMs leverage

- their training on diverse datasets,
- vector representations,
- self-attention mechanisms, and
- few-shot learning capabilities

to understand and respond effectively to questions that are similar to prompts.

how?

1
LLMs are trained on vast datasets containing various text forms, allowing them to recognize patterns in language.

When a question resembles a previously encountered prompt, the model can leverage its understanding of that context to formulate an appropriate response

2
LLMs utilize vector embeddings to represent words and phrases.

These vectors enable the model to determine semantic similarity between questions and prompts, allowing it to identify relevant examples from its training data

how?

3
The self-attention mechanism in transformers allows LLMs to focus on different parts of the input sequence while generating responses.

This capability helps the model weigh the importance of specific words in relation to others, enhancing its contextual understanding

4
LLMs can generalize from provided examples through few-shot learning, where they apply learned patterns from similar queries to new ones.
This flexibility enables them to adapt their responses based on context and prior examples

how?

5

In conversational settings, LLMs can maintain context over multiple turns, allowing them to understand follow-up questions or similar inquiries based on previous interactions

ChatGPT Training Process Explained

STEP 1

Prompt

Labeler demonstrates the designed output

Dataset for supervised learning

STF Model

STEP 2

Prompt

STF Model

Answer 1 | Answer 2 | Answer 3 | Answer 4

Labeler ranks the output

Dataset for training the Reward Model

Reward Model

When SFT can't improve with current RM, repeat Step 2 to update RM

STEP 3

Use updated RM to continue training STF Model

Prompt

STF Model

Answer

Reward Model

Reward

Update SFT Model

SFT = supervised fine-tuning

AGI

what is AGI? = Artificial General Intelligence

surpasses human cognitive capabilities across a wide range of tasks

Artificial General Intelligence (AGI) is the goal of AI that would allow machines to think and learn like humans

good vs not so good

non-deterministic

learn over time with examples

watch chess matches

deterministic

A deterministic algorithm produces the same output every time it is run with a specific input. The process it follows is predictable and can be replicated.

think chess moves

Thank you!

any questions?

videos –explained simply

transformers

https://www.youtube.com/watch?v=zxQyTK8quyY

decoder-only

https://www.youtube.com/watch?v=bQ5BoolX9Ag

attention

https://www.youtube.com/watch?v=PSs6nxngL6k

backpropagation

https://www.youtube.com/watch?v=IN2XmBhILt4

https://www.youtube.com/watch?v=iyn2zdALii8

https://www.youtube.com/watch?v=GKZoOHXGcLo