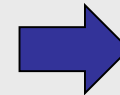
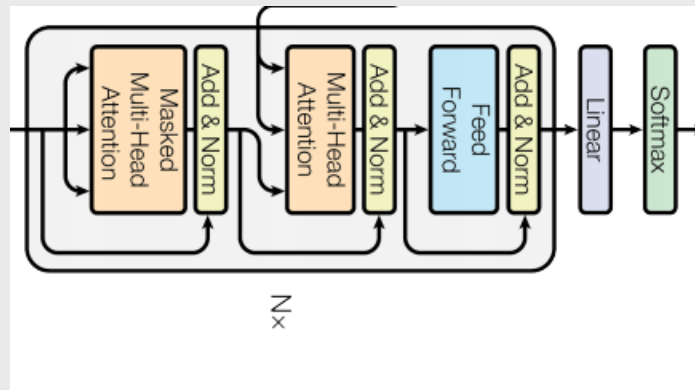
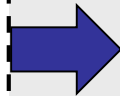


Sentiment Analysis with Neural Network Transformers



Sentiment = 0.5

Sentiment

Sentiment

- **Tweet 1: My birthday cake was awful**

Sentiment

- **Tweet 1: My birthday cake was awful**
- **Tweet 2: My birthday cake was great**

Sentiment

- **Sentiment is conveyed by specific words**

Sentiment

- **Sentiment is conveyed by specific words**
- **Maybe we could use a word frequency approach to measure sentiment**

Sentiment

- **Sentiment is conveyed by specific words**
- **Maybe we could use a word frequency approach to measure sentiment**
- **Naïve Bayes classifier – measure sentiment using term frequency embeddings**

Sentiment and Context

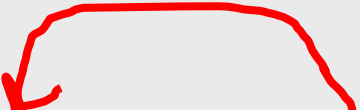
Sentiment and Context

- **Tweet 1: My birthday cake was great, if you want my honest opinion**


Sentiment and Context

- **Tweet 1: My birthday cake was great, if you want my honest opinion**
- **Tweet 2: My birthday cake was great, if you want me to get diabetes**

Sentiment and Context

- **Tweet 1: My birthday cake was great, if you want my honest opinion**
 - **Tweet 2: My birthday cake was great, if you want me to get diabetes**
- 

Sentiment and Context

- **Tweet 1: My birthday cake was great, if you want my honest opinion**
 - **Tweet 2: My birthday cake was great, if you want me to get diabetes**
- 
- Two red arrows originate from the end of the first tweet and point to the words 'great' and 'diabetes' in the second tweet, illustrating how the context of the first tweet influences the sentiment of the second.

Sentiment and Context

- **Sentiment is conveyed by specific words**

Sentiment and Context

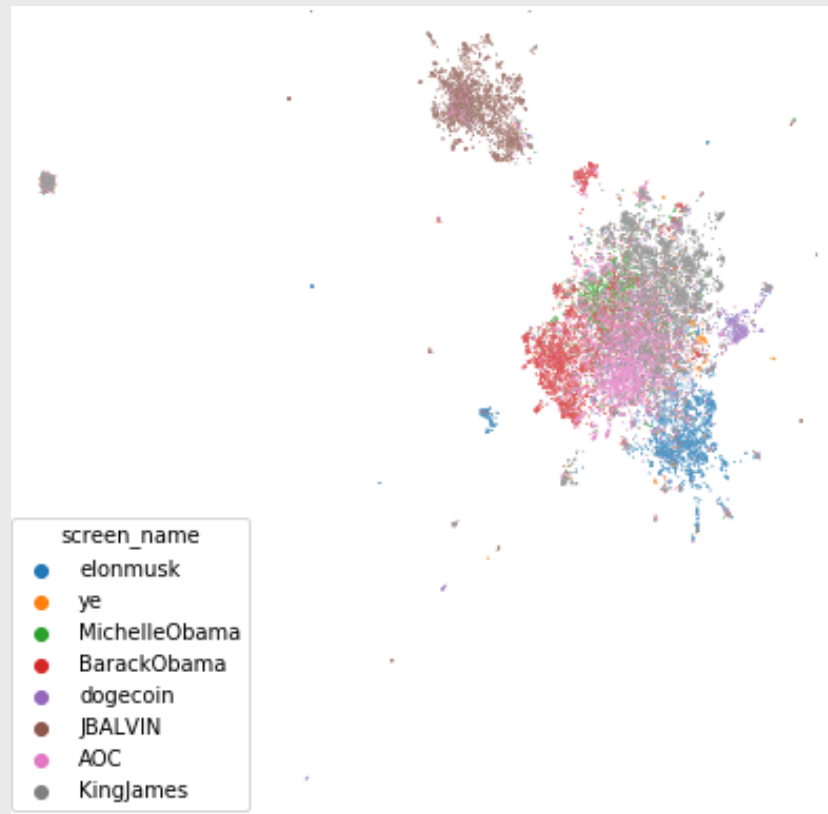
- Sentiment is conveyed by specific words
- We also need to know the **context** of the words

Sentiment and Context

- Sentiment is conveyed by specific words
- We also need to know the **context** of the words
- Context = which words pay **attention** to which words

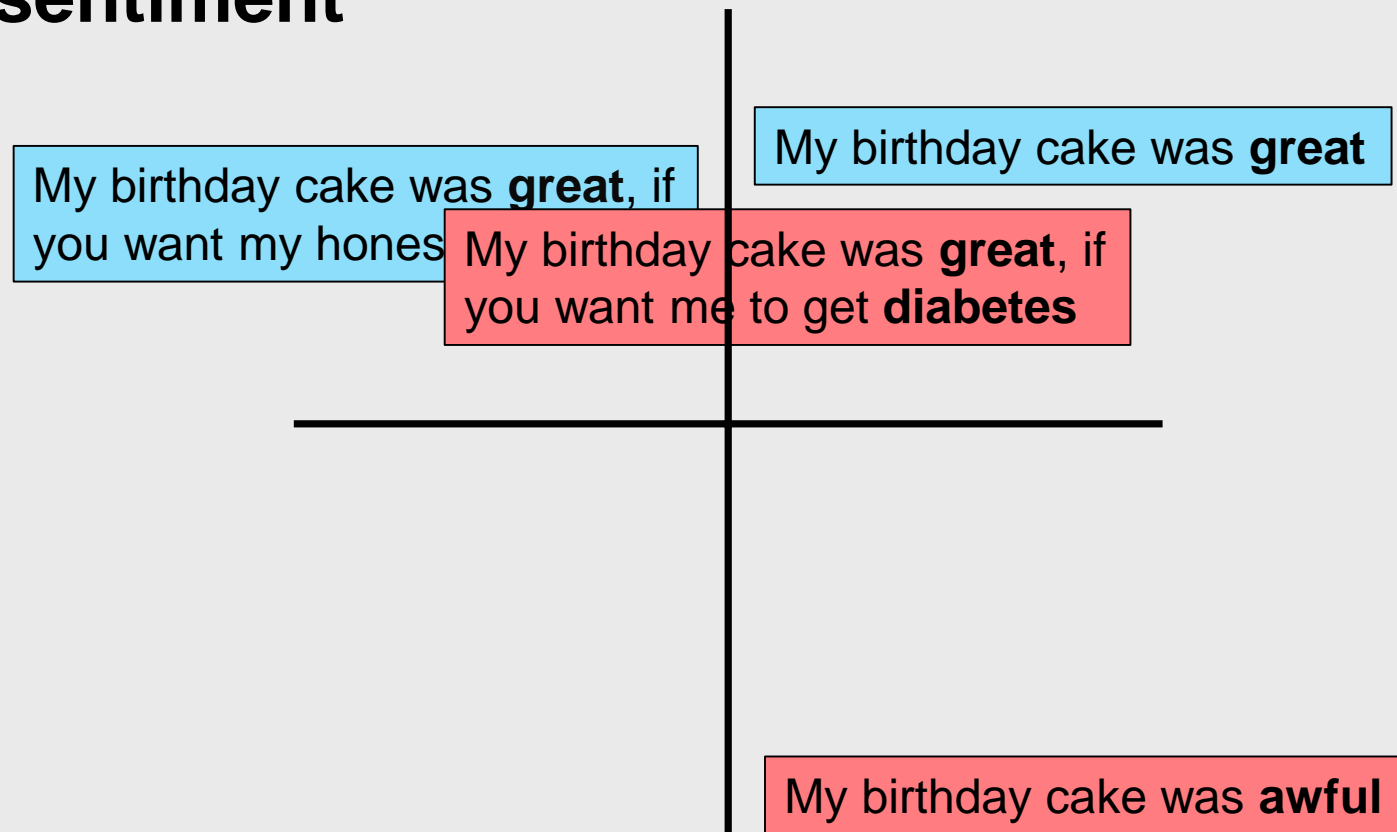
Embeddings

- We have seen how embeddings make clustering text easy



Context Dependent Embeddings

- A clustering type of embedding may cluster tweets with similar words, but different sentiment



Context Dependent Embeddings

- Context dependent embedding can cluster by sentiment



Context Dependent Embeddings

Context Dependent Embeddings

- We need a model that allows words in a sentence to pay “attention” to other words

Context Dependent Embeddings

- **We need a model that allows words in a sentence to pay “attention” to other words**
- **Words can pay attention in different ways**

Context Dependent Embeddings

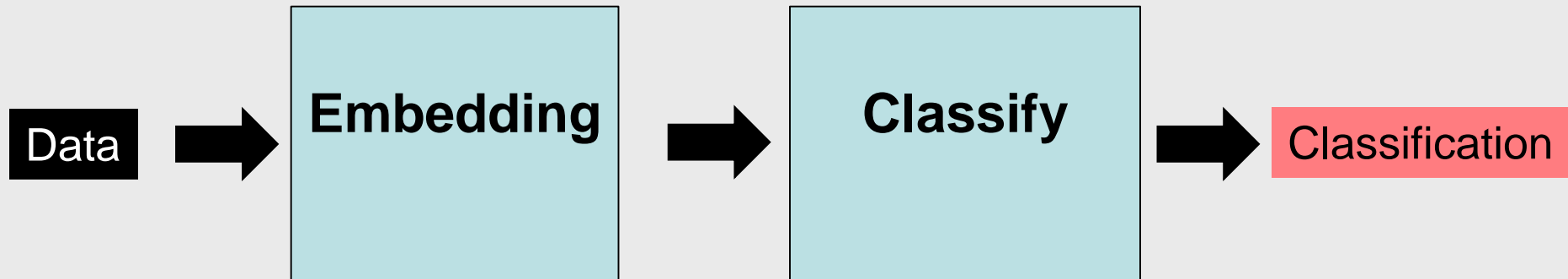
- **We need a model that allows words in a sentence to pay “attention” to other words**
- **Words can pay attention in different ways**
- **We can choose the type of “attention” that captures sentiment**

Context Dependent Embeddings

- We need a model that allows words in a sentence to pay “attention” to other words
- Words can pay attention in different ways
- We can choose the type of “attention” that captures sentiment
- Solution: **Neural Network Transformers**

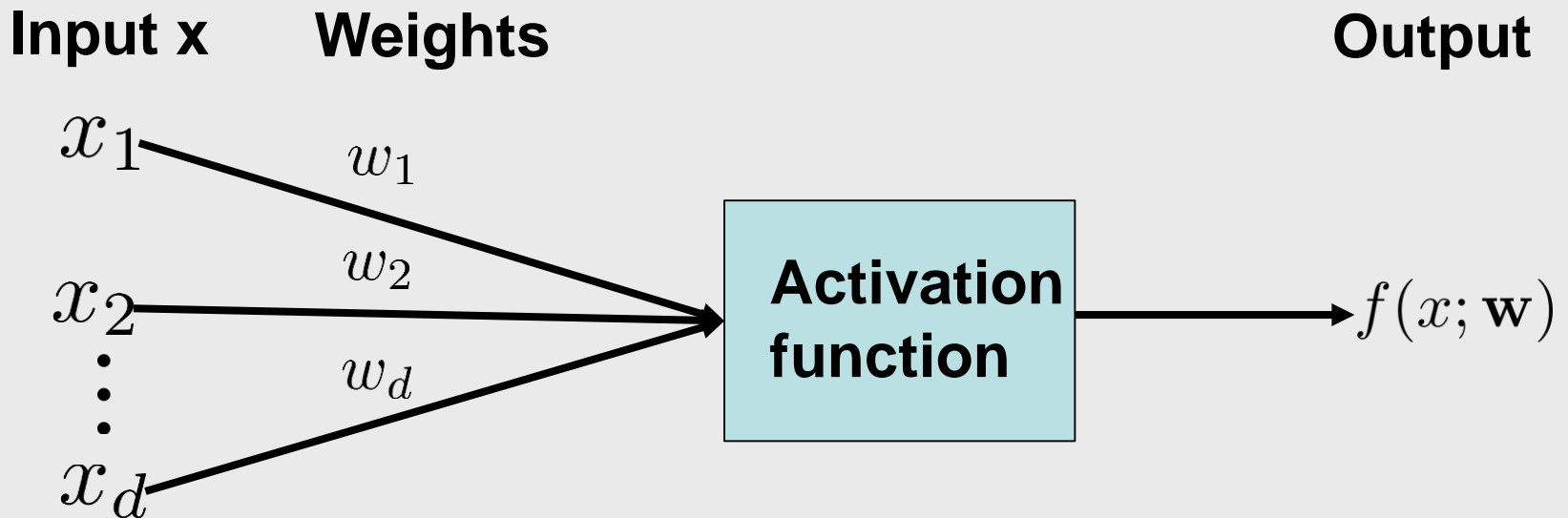
Neural Networks

- Neural networks let you learn very complex embeddings
- You can classify data using these embeddings



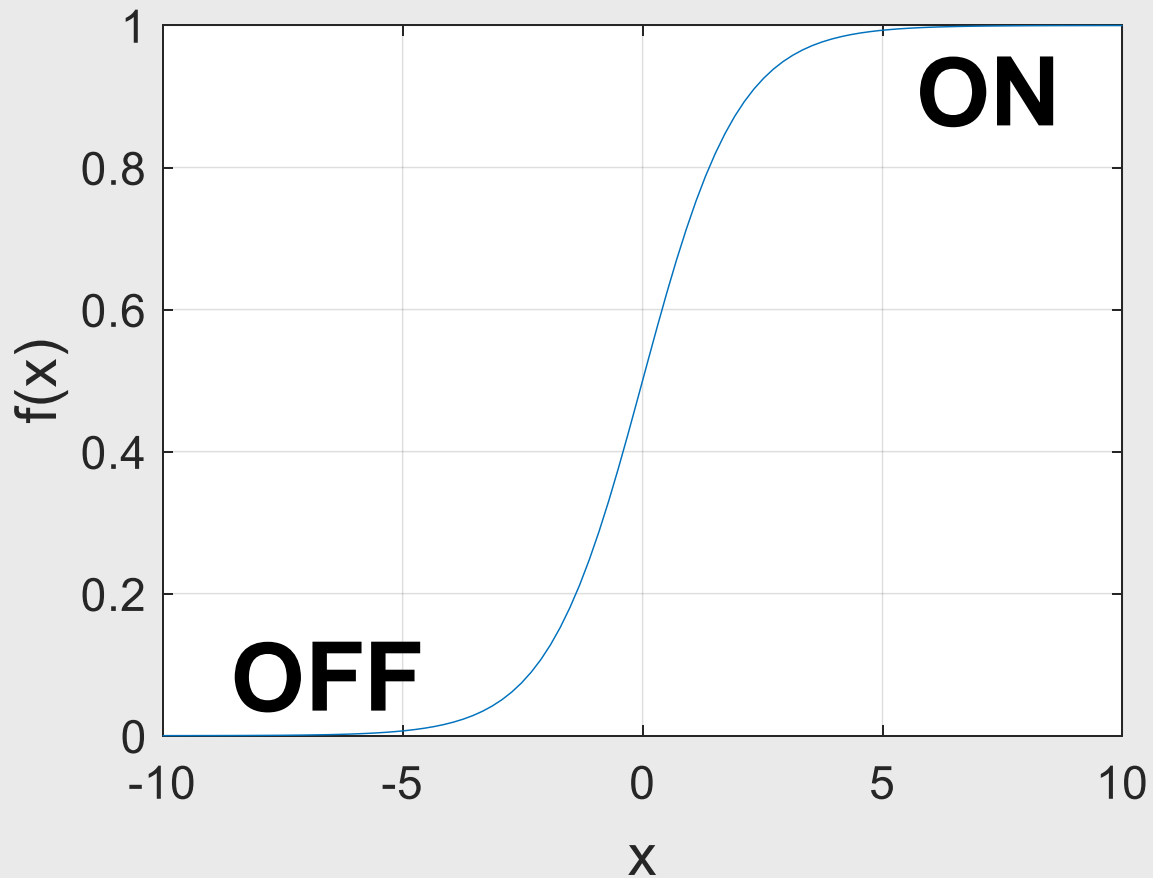
Neurons

- Neurons are the core building block of a neural network is the neuron



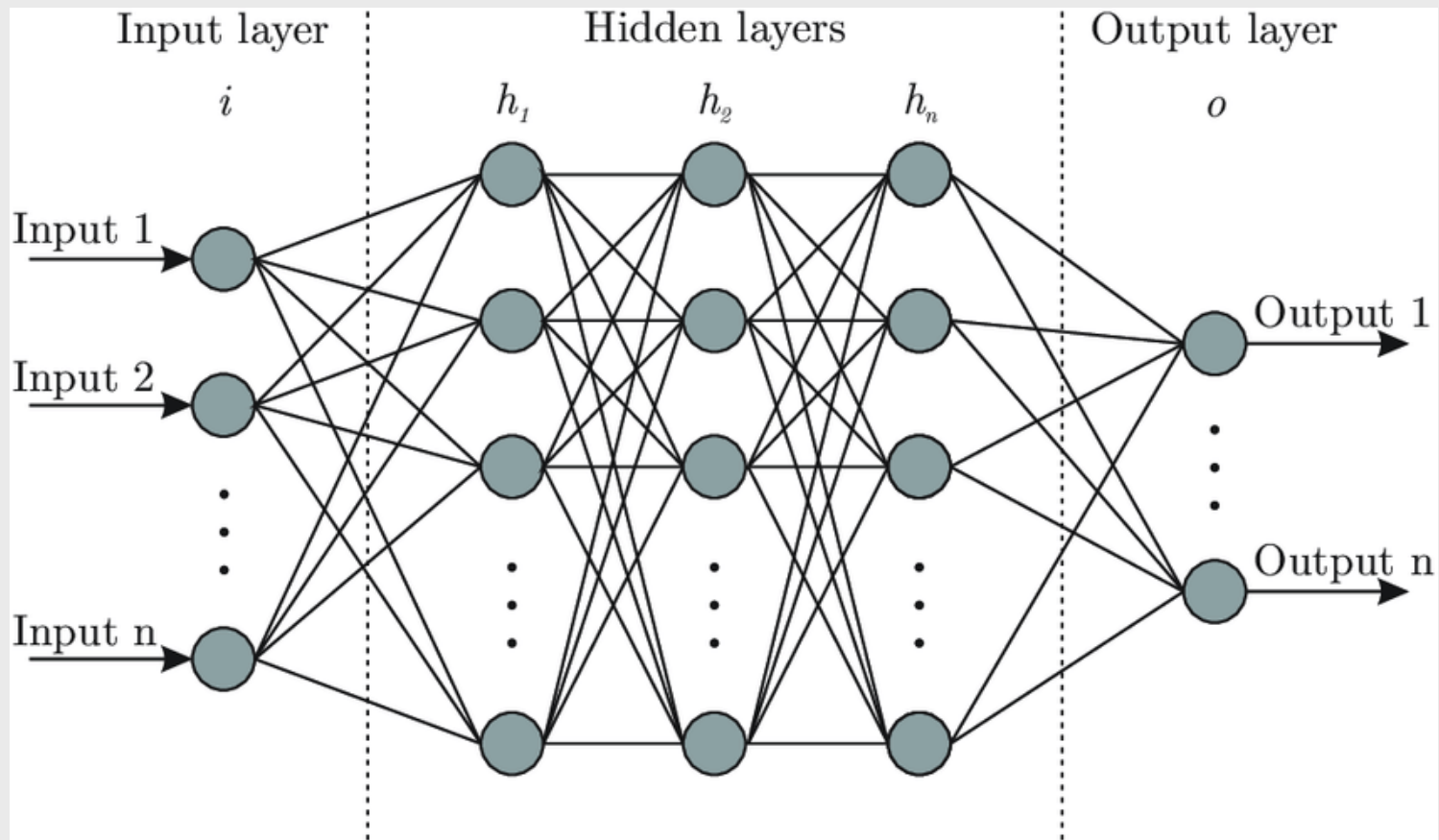
Activation Function

- Common activation function - sigmoid



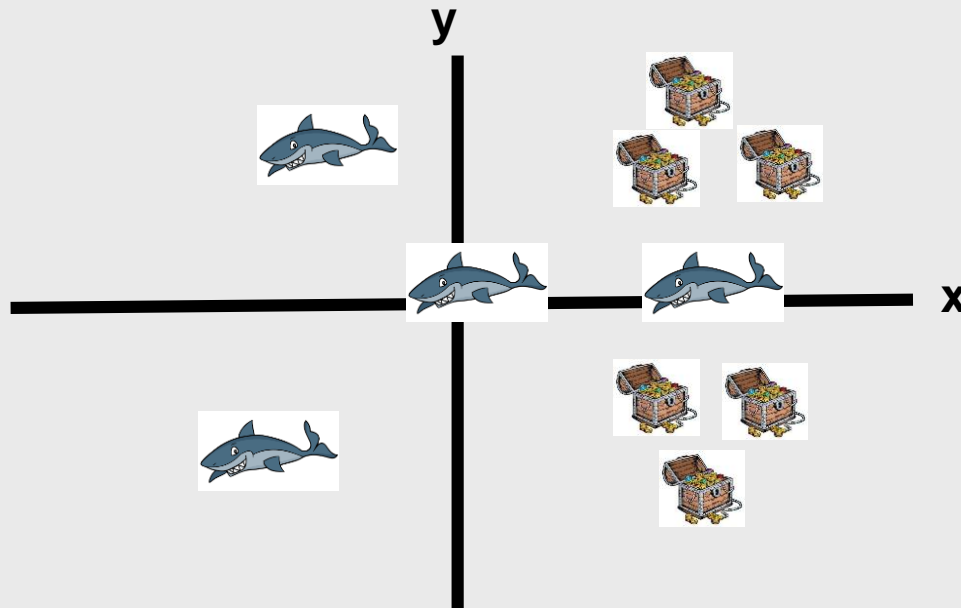
Deep Neural Network

- We can have multiple layers of neurons



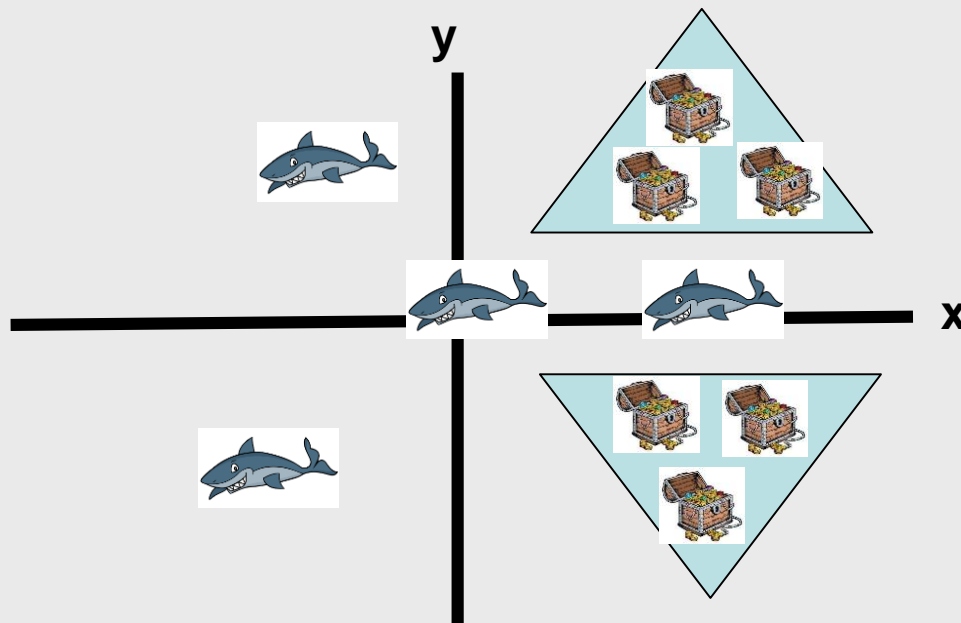
Classification Problem

- Build a neural network to classify these points



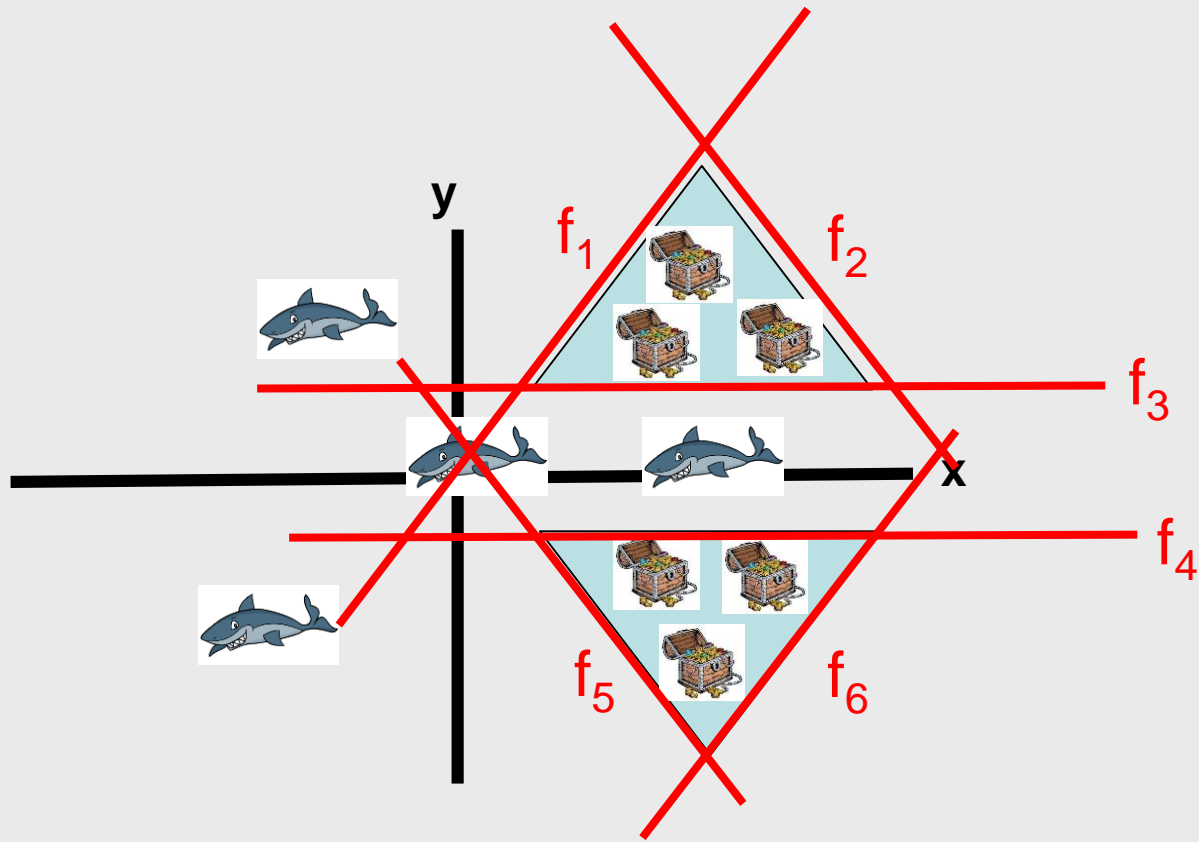
Classification Problem

- 2 classification regions – the 2 triangles



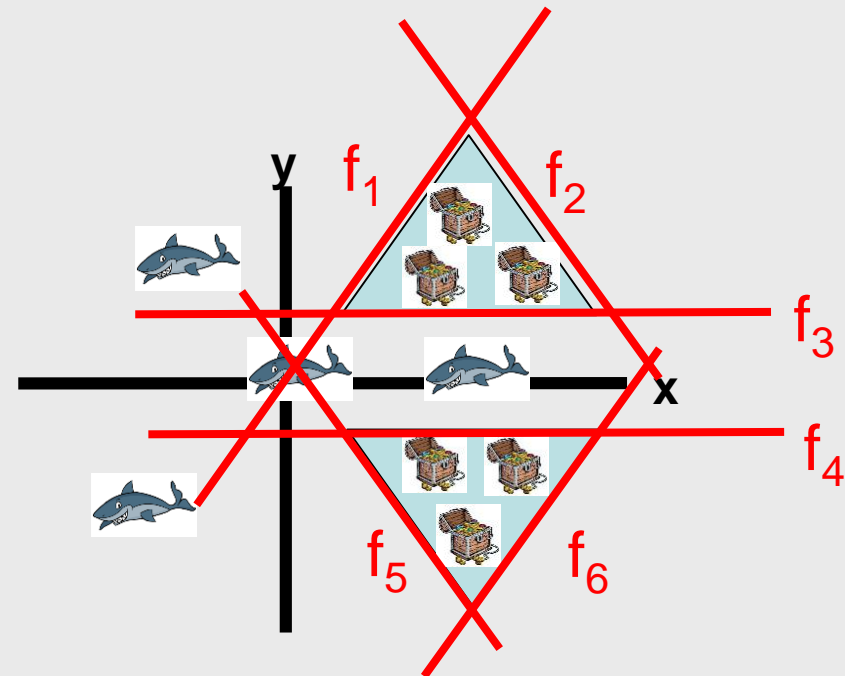
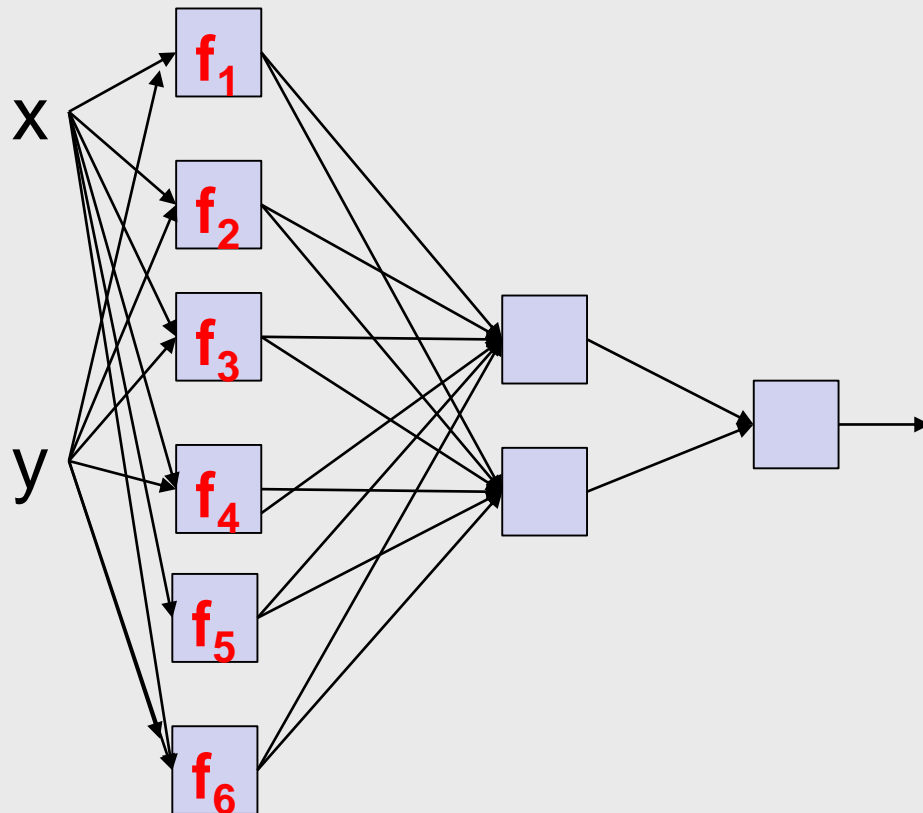
Neural Network Solution

- 2 classification regions – the 2 triangles
- 6 features – one for each side of the triangles



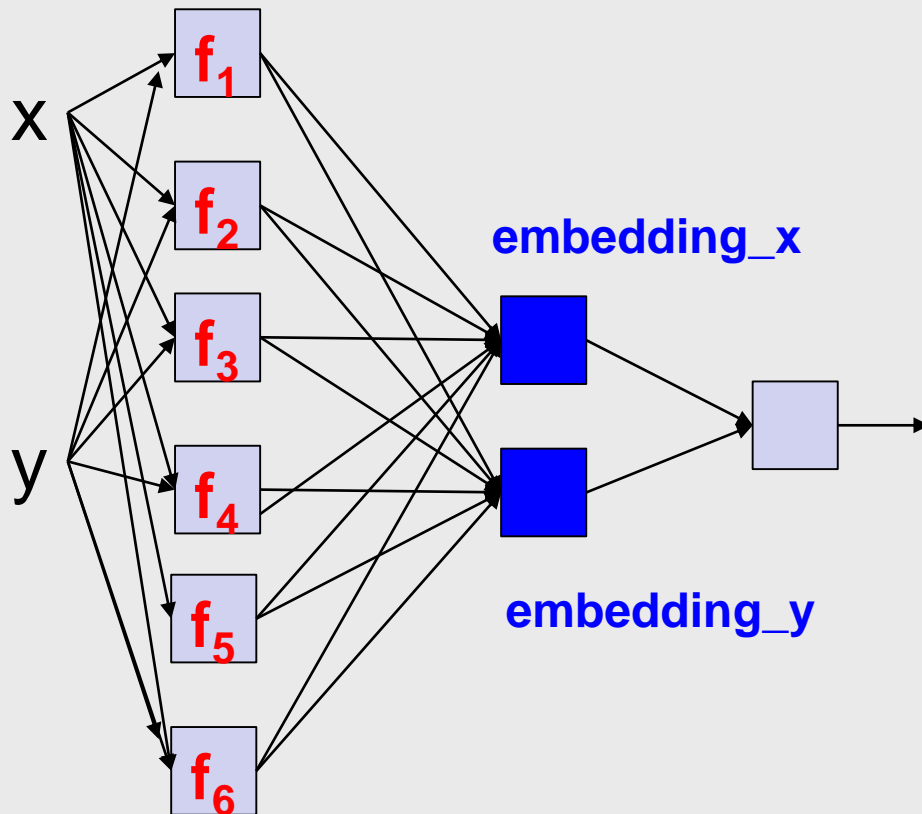
Neural Network Solution

- Classify using a neural network



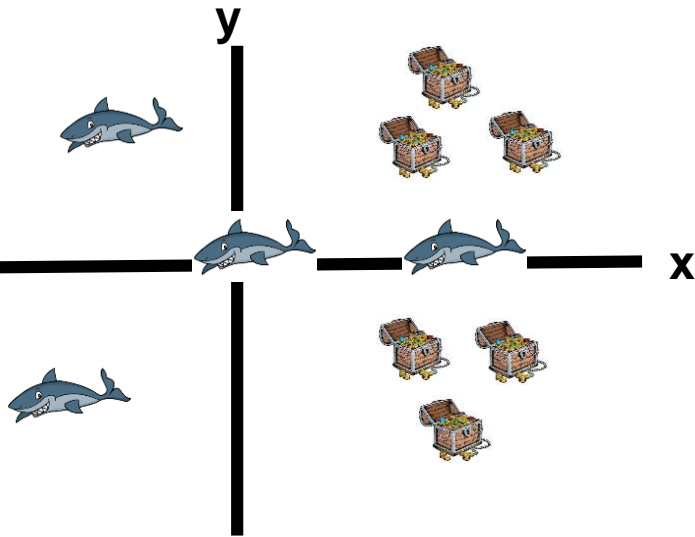
Neural Network Embedding

- Hidden layer can act as an embedding



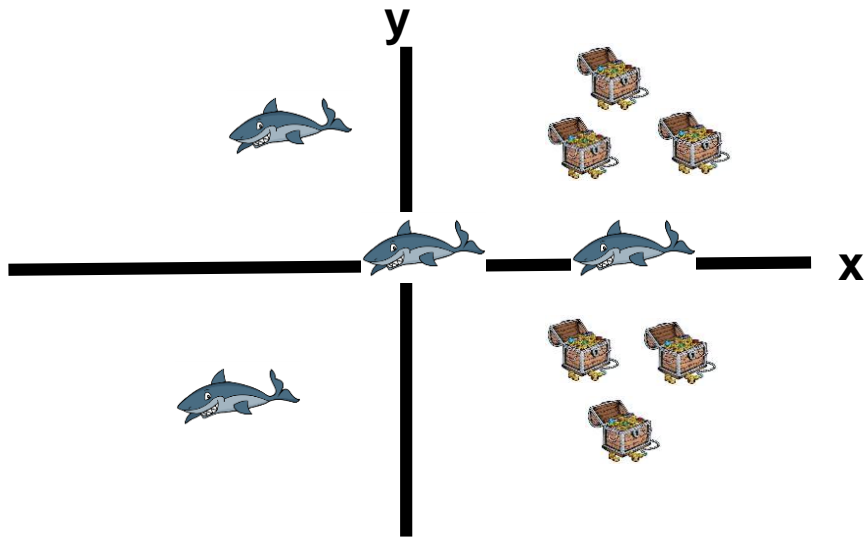
Neural Network Embedding

Initial Embedding

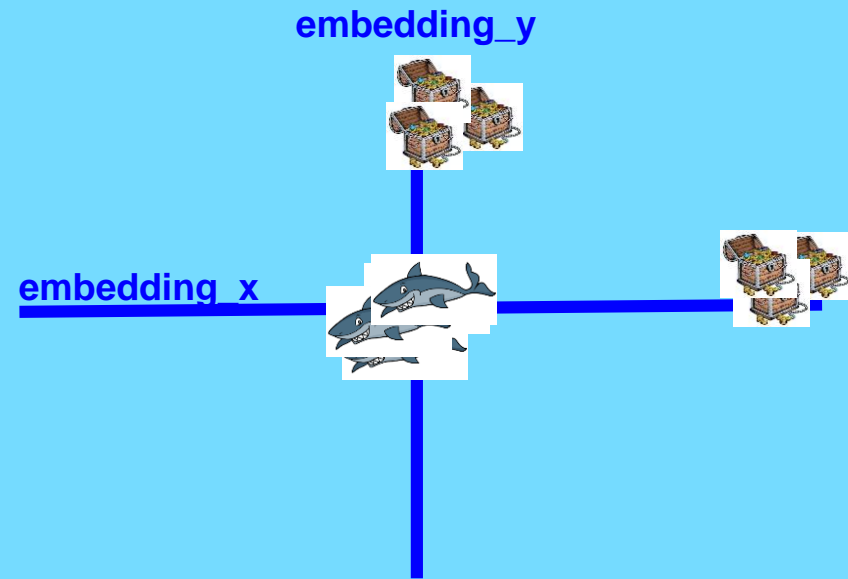


Neural Network Embedding

Initial Embedding



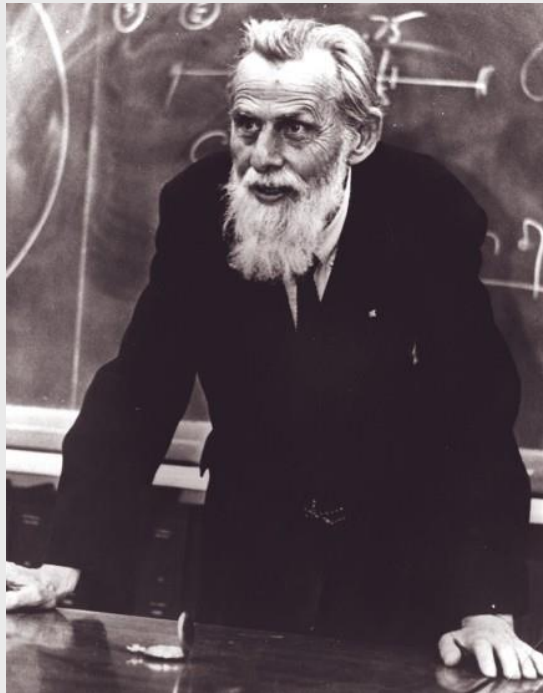
Neural Network Embedding



NEURAL NETWORK HISTORY

Origins of Neural Networks

- 1943 – Walter Pitts and Warren McCullough propose “nervous nets”



Haters

- **1969 – Marvin Minsky and Seymour Papert say single layer neural network cant do that much AND computers are too slow to use neural nets**

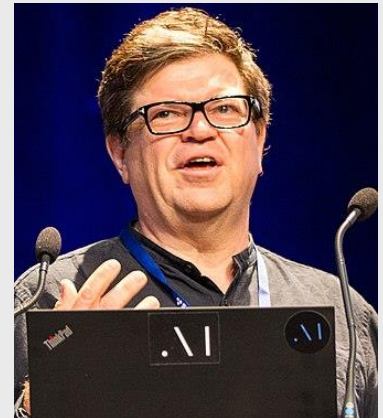


Haters



Neural Nets in the 1980s

- **1986 – Geoffrey Hinton and co-authors use back-propagation to train a neural network**
- **1989 – Yann LeCun creates convolutional neural networks to read zipcodes**

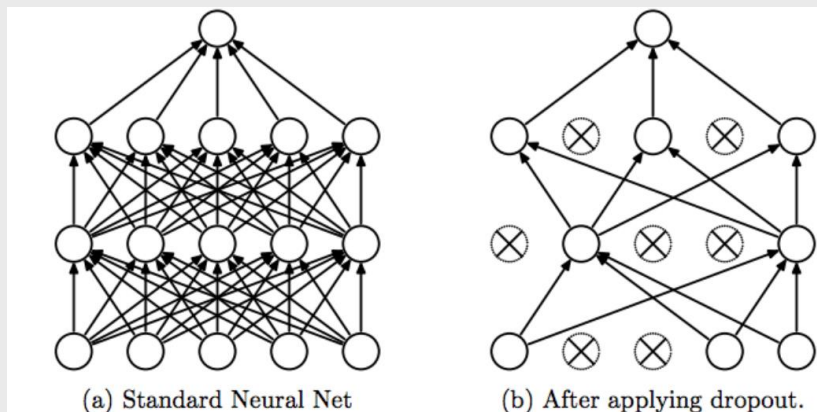


Neural Nets in the 2000s



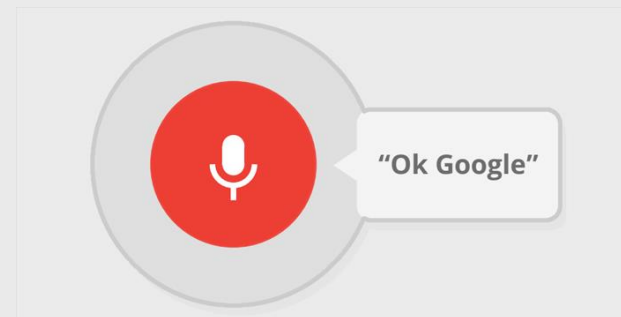
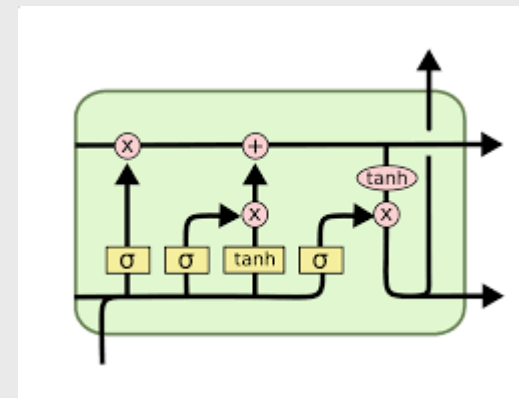
The Beginning of Deep Learning

- **2012 – Hinton and his students win a drug discovery contest held by Merck – using a deep neural network trained using the “dropout” technique he invented**



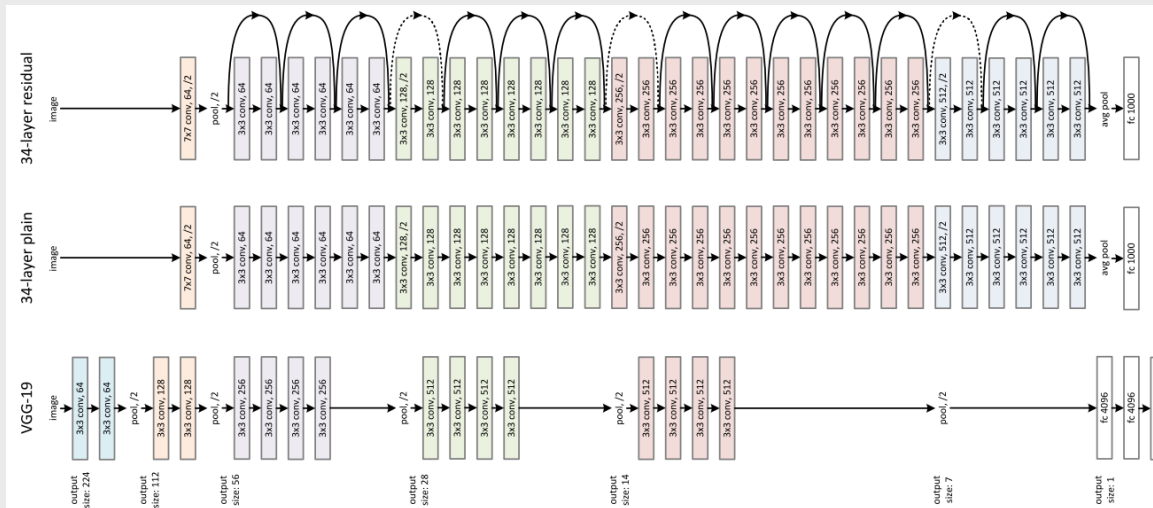
Long-term Short-Term Recurrent Neural Networks

- 1987 – Long-term short-term recurrent neural networks (LSTM RNN) invented
- 2010s – breakthroughs in speech recognition achieved using LSTM RNNs
- Google voice search uses LSTM RNNs

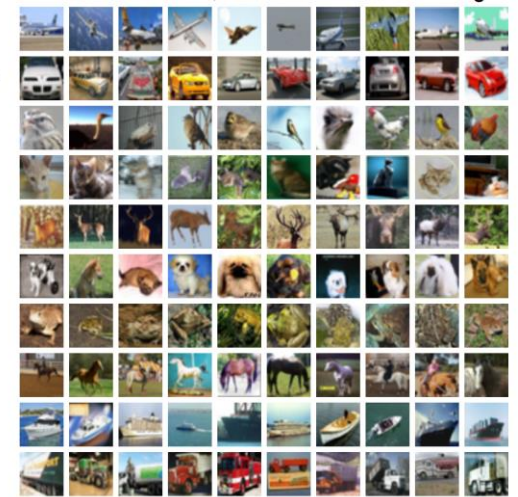


Residual Nets

- 2015 – Residual Networks (ResNets) proposed
- Revolutionizes object recognition
- Error rates near 5%



airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck



Transformers

- Developed in 2017 by Google
- Revolutionized natural language processing

Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Lukasz Kaiser*
Google Brain
lukaszkaizer@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention

What Can Transformers Do?

- **Measure sentiment**
- **Translation**
- **Web search**
- **Text summarization**
- **Question answering**
- **Generate text**

Transformer Architecture

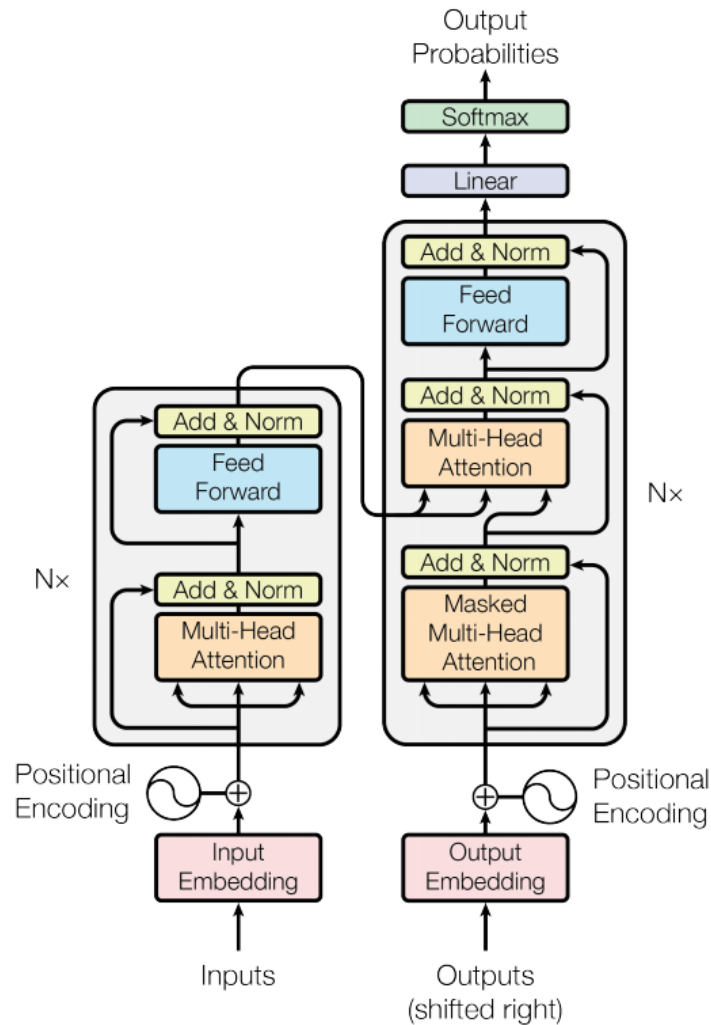
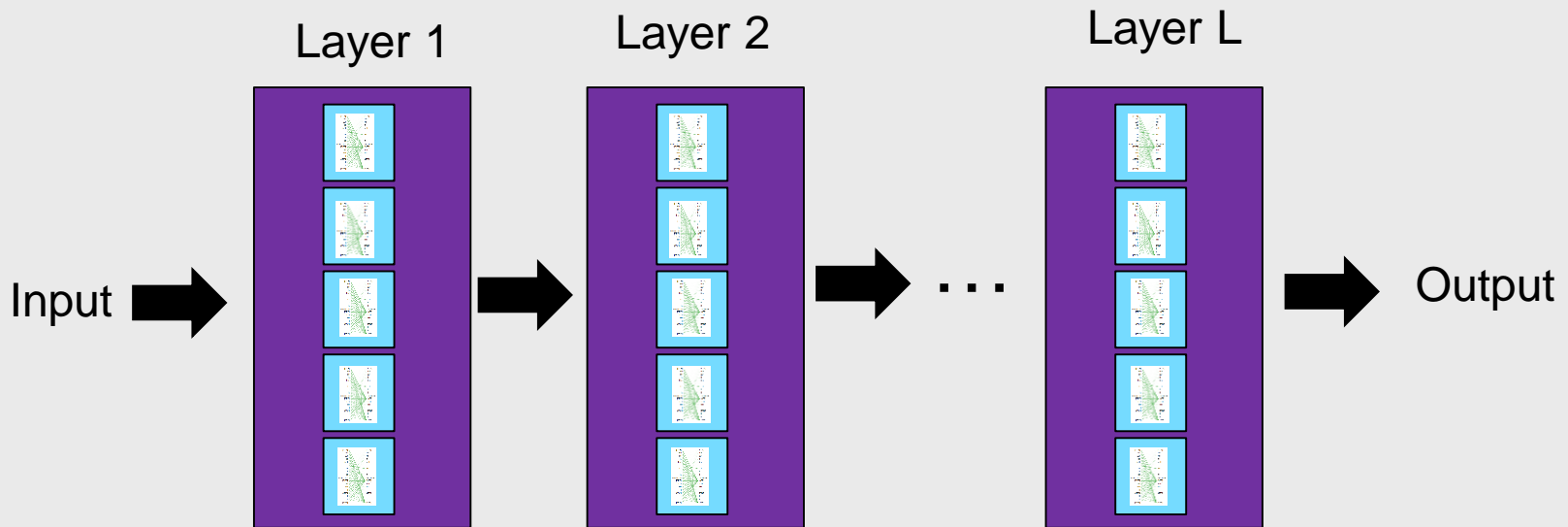


Figure 1: The Transformer - model architecture.

Transformer Architecture

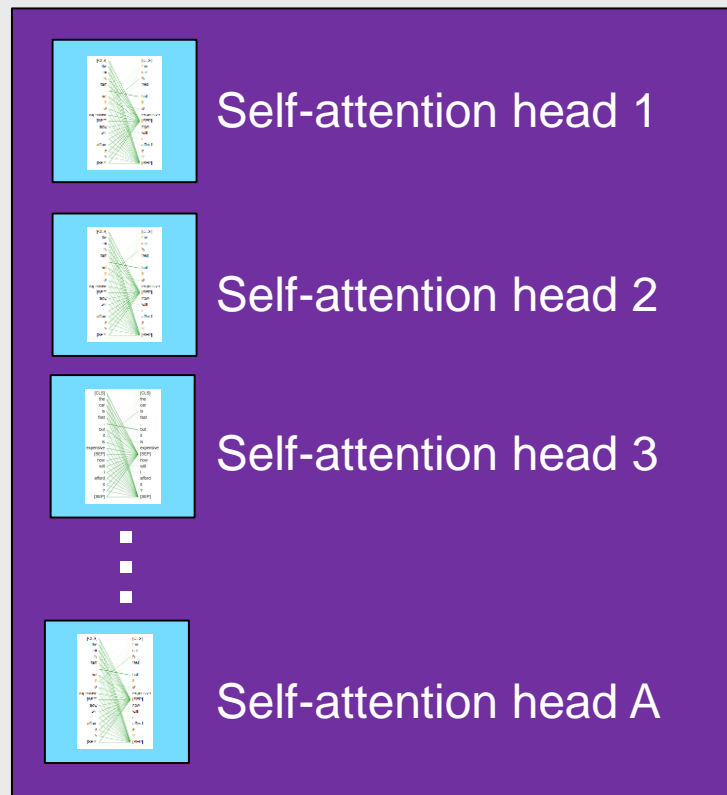
- The transformer has many layers



Transformer Layers

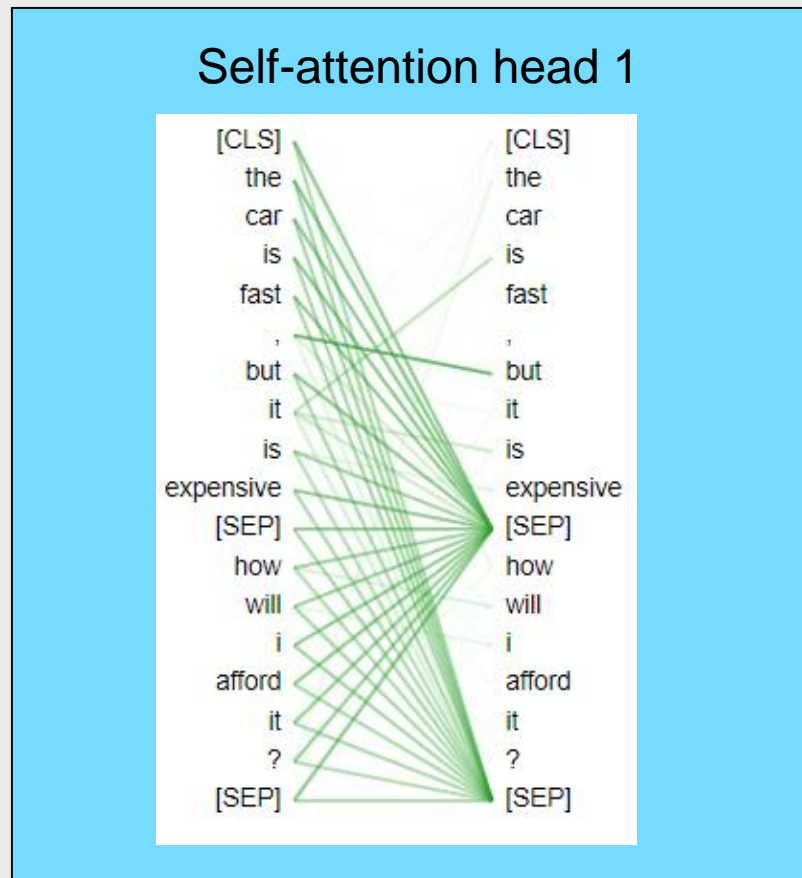
- Each layer has many self-attention heads

Layer 1

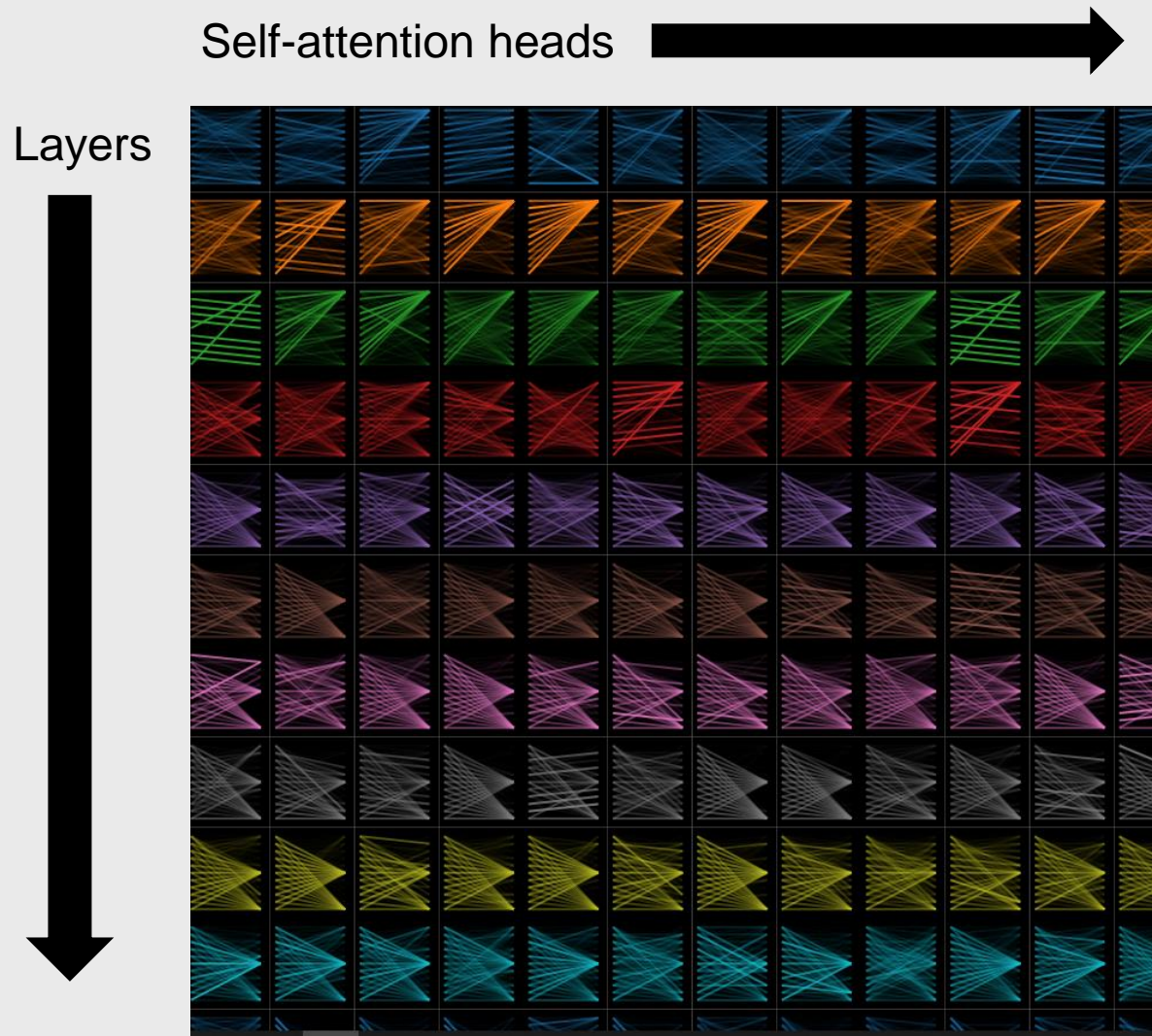


Self-Attention Head

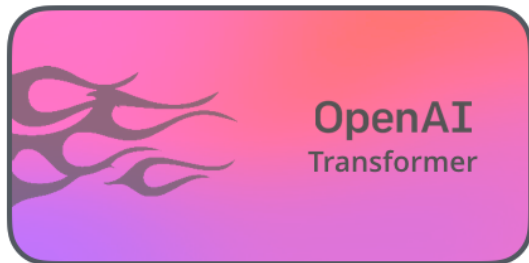
- Each self-attention head contains attention weights from each word to each other word



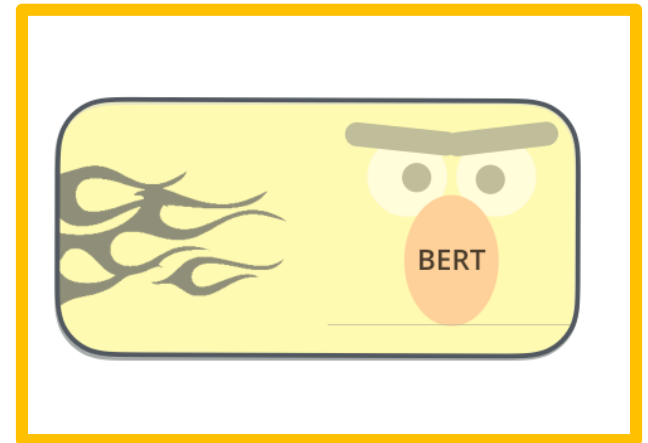
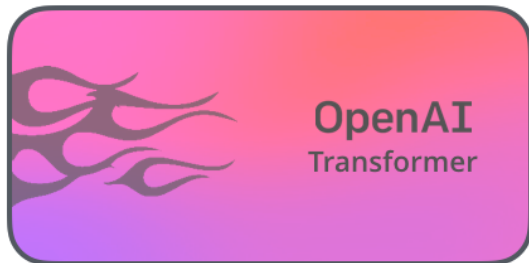
Visualizing the Brain of a Transformer



Popular Transformers



Popular Transformers



BERT



- **BERT = Bi-directional Encoder Representations From Transformers**
- **Released in 2018 by Google**
- **Base BERT has 100 million parameters**
 - 12 layers
 - 12 attention heads
 - 768 dimensional word embedding
- **Trained on books and Wikipedia (3.3 billion words)**

Training BERT

1 - **Semi-supervised** training on large amounts of text (books, wikipedia..etc).

The model is trained on a certain task that enables it to grasp patterns in language. By the end of the training process, BERT has language-processing abilities capable of empowering many models we later need to build and train in a supervised way.

Semi-supervised Learning Step

Model:



Dataset:



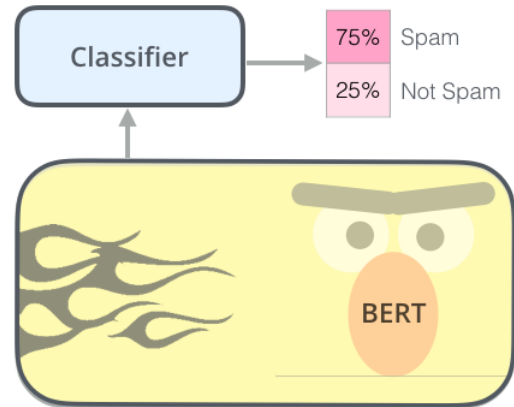
Objective:

Predict the masked word
(language modeling)

2 - **Supervised** training on a specific task with a labeled dataset.

Supervised Learning Step

Model:
(pre-trained
in step #1)



Dataset:

Email message	Class
Buy these pills	Spam
Win cash prizes	Spam
Dear Mr. Atreides, please find attached...	Not Spam

Masked Language Model Task

- BERT is trained to learn a masked language model
 - Guess **[MASK]** words in a sentence

Data	Prediction
I went to the [MASK] to buy milk.	[MASK] = store
I graduated from [MASK] and got a degree.	[MASK] = college
I had a [MASK] and it tasted [MASK] !	[MASK] = hamburger [MASK] = amazing

Pre-Trained Transformers: Hugging Face



Hugging Face

Search models, datasets, users...

Models

Datasets

Pricing

Resources

Log In

Sign Up

Tasks

- Fill-Mask
- Question Answering
- Summarization
- Table Question Answering
- Text Classification
- Text Generation
- Text2Text Generation
- Token Classification
- Translation
- Zero-Shot Classification
- + 4

Libraries

- PyTorch
- TensorFlow
- + 9

Datasets

- wikipedia
- squad
- c4
- bookcorpus
- dcep europarl jrc-acquis
- CLUECorpusSmall
- oscar
- squad_v2
- + 205

Languages

- en
- es
- fr
- sv
- fi
- de
- multilingual
- zh
- + 329

Licenses

- apache-2.0
- mit
- gpl-3.0
- + 13

Models 6132

Search Models

Sort: Most Downloads

bert-base-uncased

Fill-Mask • Updated Dec 11, 2020 • 22,766k

cl-tohoku/bert-base-japanese-whole-word-masking

Fill-Mask • Updated Jan 25 • 4,354k

xlm-roberta-base

Fill-Mask • Updated Dec 11, 2020 • 2,576k

bert-large-uncased

Fill-Mask • Updated Jan 13 • 2,031k

bert-large-cased

Fill-Mask • Updated Jan 13 • 1,717k

gpt2

Text Generation • Updated Dec 11, 2020 • 815k

t5-small

Translation • Updated Dec 11, 2020 • 751k

sentence-transformers/distilbert-base-nli-stsb-m...

Updated Aug 31, 2020 • 714k

valhalla/t5-small-qg-hl

Text2Text Generation • Updated Dec 11, 2020 • 676k

distilbert-base-uncased

Fill-Mask • Updated Dec 11, 2020 • 11,104k

jplu/tf-xlm-roberta-base

Fill-Mask • Updated Dec 11, 2020 • 3,945k

roberta-base

Fill-Mask • Updated Dec 11, 2020 • 2,158k

bert-base-cased

Fill-Mask • Updated Dec 15, 2020 • 1,913k

valhalla/t5-small-qa-qg-hl

Text2Text Generation • Updated Dec 11, 2020 • 1,135k

distilbert-base-uncased-finetuned-sst-2-english

Text Classification • Updated Feb 9 • 814k

roberta-large

Fill-Mask • Updated Dec 11, 2020 • 721k

facebook/bart-large-mnli

Zero-Shot Classification • Updated Dec 11, 2020 • 680k

t5-base

Translation • Updated Dec 11, 2020 • 606k

Evaluating Language Models: GLUE








- **GLUE = general language understanding and evaluation**
- **GLUE is a set of benchmark tasks to evaluate language models like BERT**

GLUE Tasks

Task type	Description
Acceptability	Is the sentence grammatically correct
Sentiment	Can you predict the sentiment of the sentence
Question answering	Does the second sentence answer the question in the first sentence
Natural language inference	Does the second sentence entail the hypothesis in the first sentence
Pronoun referral	To what does the pronoun in a sentence refer
Sentence similarity	Are the two sentences paraphrases of each other






GLUE Leaders

- Human GLUE score = 87.1
- GLUE leaderboard: <https://gluebenchmark.com/leaderboard>

Rank Name		Model	URL	Score
1	ERNIE Team - Baidu	ERNIE		90.9
2	DeBERTa Team - Microsoft	DeBERTa / TuringNLRv4		90.8
3	HFL iFLYTEK	MacALBERT + DKM		90.7
+	4 Alibaba DAMO NLP	StructBERT + TAPT		90.6
+	5 PING-AN Omni-Sinitic	ALBERT + DAAF + NAS		90.6
6	T5 Team - Google	T5		90.3
7	Microsoft D365 AI & MSR AI & GATECHMT-DNN-SMART			89.9
+	8 Huawei Noah's Ark Lab	NEZHA-Large		89.8
+	9 Zihang Dai	Funnel-Transformer (Ensemble B10-10-10H1024)		89.7
+	10 ELECTRA Team	ELECTRA-Large + Standard Tricks		89.4

SuperGLUE Leaders

- Human SuperGLUE score = 89.8
- SuperGLUE leaderboard: <https://super.gluebenchmark.com/leaderboard>

Rank	Name	Model	URL	Score
+	1	DeBERTa Team - Microsoft	DeBERTa / TuringNLRv4	 90.3
+	2	Zirui Wang	T5 + Meena, Single Model (Meena Team - Google Brain)	90.2
	3	SuperGLUE Human Baselines	SuperGLUE Human Baselines	 89.8
+	4	T5 Team - Google	T5	 89.3
+	5	Huawei Noah's Ark Lab	NEZHA-Plus	 86.7
+	6	Alibaba PAI&ICBU	PAI Albert	86.1
+	7	Tencent Jarvis Lab	RoBERTa (ensemble)	85.9
+	8	Infosys : DAWN : AI Research	RoBERTa-iCETS	85.8
	9	Zhuiyi Technology	RoBERTa-mtl-adv	85.7
	10	Facebook AI	RoBERTa	 84.6

SuperGLUE Leaders

- Human SuperGLUE score = 89.8
- SuperGLUE leaderboard: <https://super.gluebenchmark.com/leaderboard>

Rank	Name	Model	URL	Score
+	1	DeBERTa Team - Microsoft	DeBERTa / TuringNLRv4	90.3
+	2	Zirui Wang	T5 + Meena, Single Model (Meena Team - Google Brain)	90.2
	3	SuperGLUE Human Baselines	SuperGLUE Human Baselines	89.8
+	4	T5 Team - Google	T5	89.3
+	5	Huawei Noah's Ark Lab	NEZHA-Plus	86.7
+	6	Alibaba PAI&ICBU	PAI Albert	86.1
+	7	Tencent Jarvis Lab	RoBERTa (ensemble)	85.9
+	8	Infosys : DAWN : AI Research	RoBERTa-iCETS	85.8
	9	Zhuiyi Technology	RoBERTa-mtl-adv	85.7
	10	Facebook AI	RoBERTa	84.6

Next Time: Coding Session

- **Use a pre-trained BERT sentiment classifier to measure tweet sentiment**
- **Learn how to use any model in the huggingface library**
- **Need to install some neural network packages – please try this before class because there might be issues**