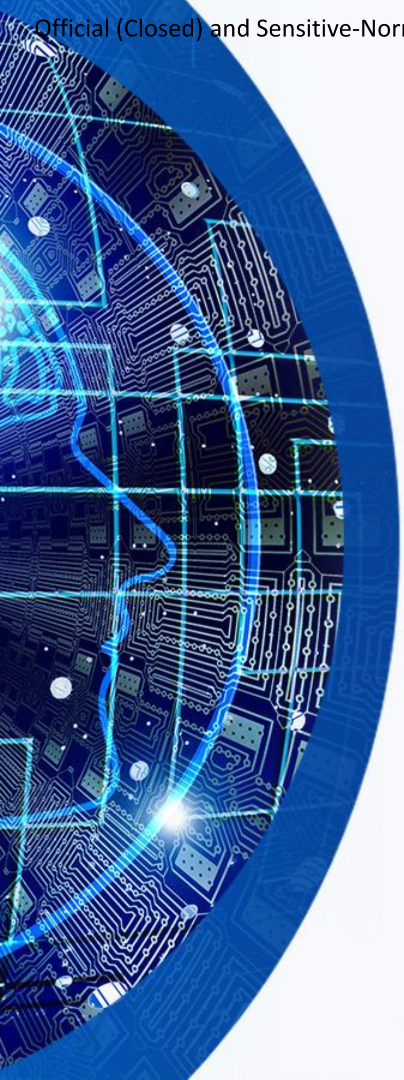
Introduction to Artificial Intelligence and Machine Learning

School of Engineering
Nanyang Polytechnic

3 What is Deep Learning



History of Deep Learning Milestones and Tools

DL Milestones

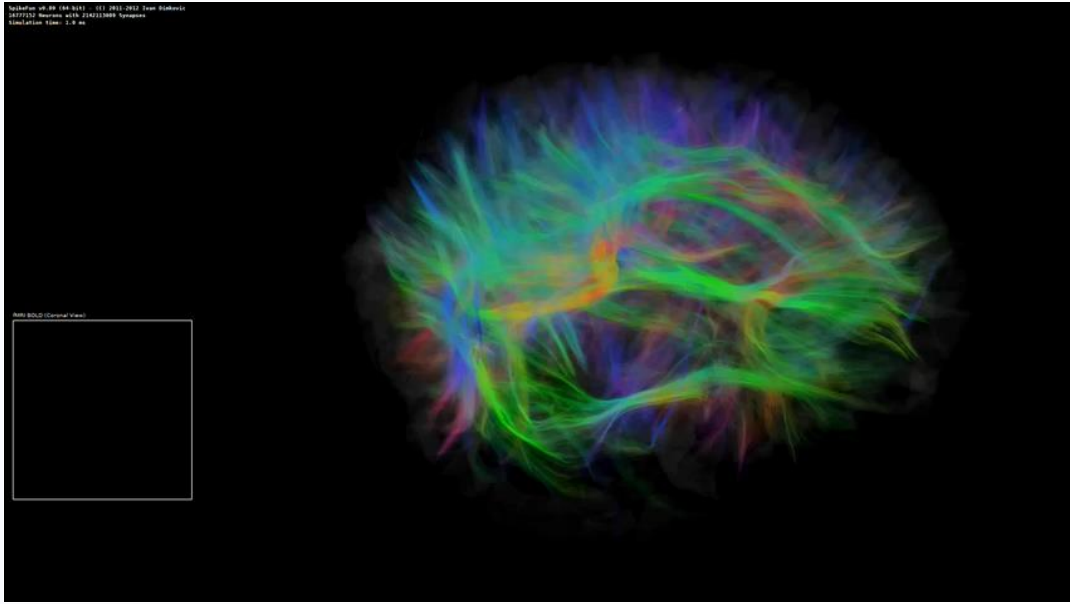
- 1943: Neural networks
- 1957: Perceptron
- 1974-86: Backpropagation, RBM, RNN
- 1989-98: CNN, MNIST, LSTM, Bidirectional RNN
- 2006: “Deep Learning”, DBN
- 2009: ImageNet
- 2012: AlexNet, Dropout
- 2014: GANs
- 2014: DeepFace
- 2015: RestNet-152
- 2016: AlphaGo
- 2017: AlphaZero, Capsule Networks
- 2018: Google BERT, OpenAI GPT
- 2019: GPT-2
- 2020: GPT-3
- 2021: GPT-3.5

DL Tools

- Mark 1 Perceptron–1960
- Torch –2002
- CUDA –2007
- Theano –2008
- Caffe –2014
- DistBelief–2011
- TensorFlow 0.1 –2015
- PyTorch0.1 –2017
- TensorFlow 1.0 –2017
- PyTorch1.0 –2017
- TensorFlow 2.0 –2019
- PyTorch1.10 - 2021
- TensorFlow 2.7 –2021

Full list of acronym can be found in the reference slides

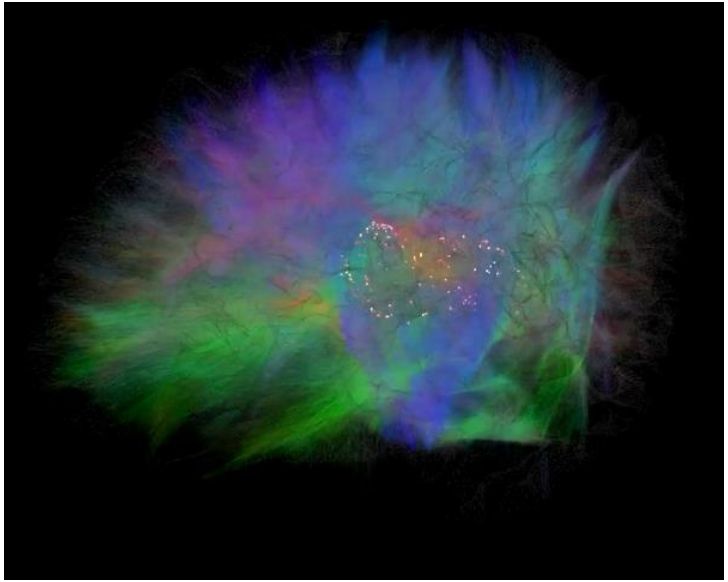
The Human Brain and the Neurons



Visualized here are 3% of the neurons and 0.0001% of the synapses in the brain.

Artificial Brain Simulation - Thalamocortical System, 16.7 Million Neurons - 2.1 Billion Synapses
https://www.youtube.com/watch?v=PM_gTOm9fgk

Biological and Artificial Neural Networks



Human Brain

- 100 billion neurons and 1,000 trillion synapses

Artificial Neural Network

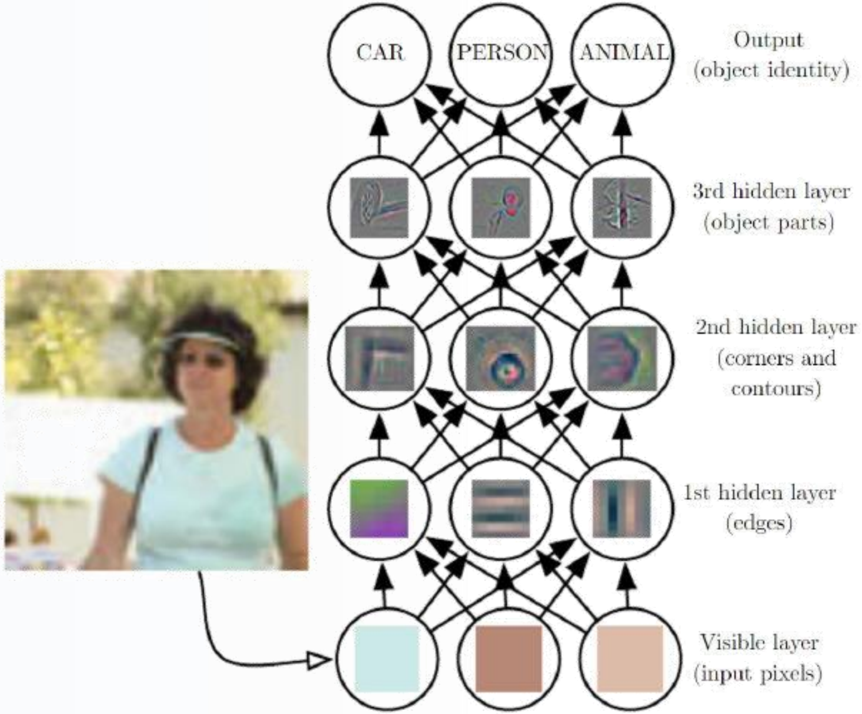
- ResNet-152: 60 million synapses
- ChatGPT-2: 50 billion neurons
- ChatGPT-3: 60-80 billion neurons and around 100 trillion synapses

Human brains have ~10,000,000 times synapses than artificial neural networks

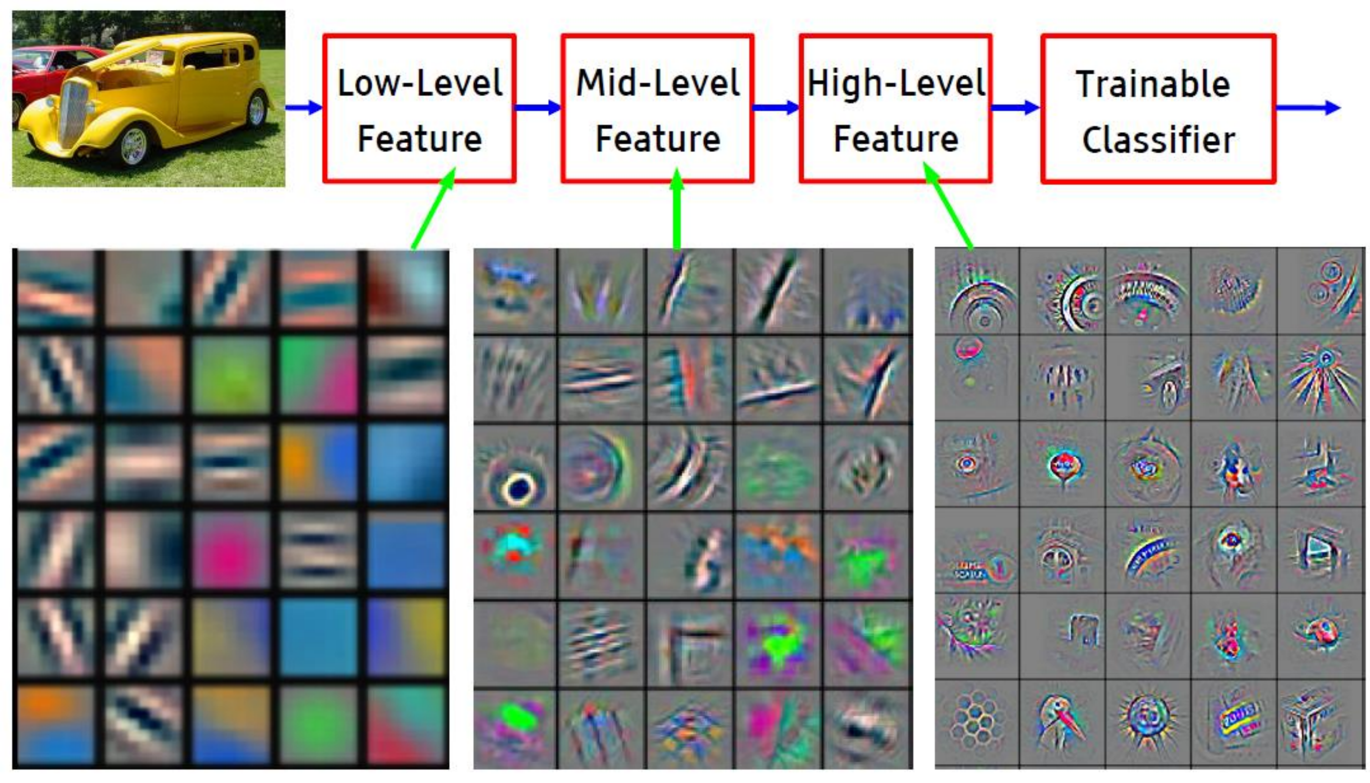
LLM Competition and Race

Year	Model	Developer	Key Features	No. of Parameters
2022	GPT-3	OpenAI	175 billion parameters, few-shot learning capabilities	175 billion
2022	LaMDA	Google	Specialized for dialogue, safety and factuality focus	137 billion
2023	GPT-4	OpenAI	Improved context understanding, multimodal capabilities	1 trillion
2023	PaLM 2	Google	Enhanced multilingual and reasoning abilities	540 billion
2023	Claude	Anthropic	Safety and alignment-focused LLM	52 billion
2023	LLaMA (LLaMA 2)	Meta	Open-source model with various sizes up to 70 billion parameters	70 billion
2024	Gemini 1	Google DeepMind	Integration of advanced reasoning and problem-solving skills	Not publicly disclosed
2024	Mistral	Mistral AI	Compact, efficient model with high performance	7 billion

Deep Learning is Representation Learning (aka feature learning)

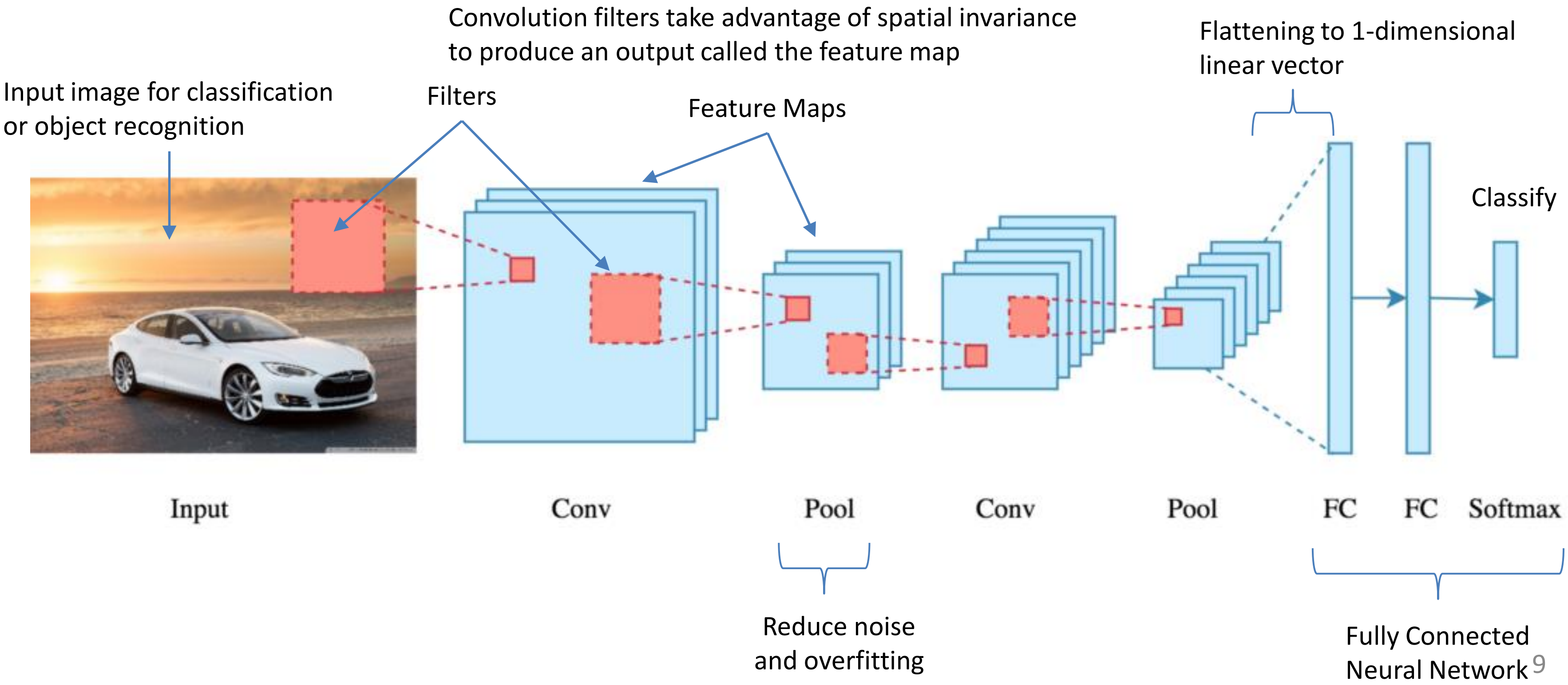


Deep Learning = Learning Hierarchical Representations

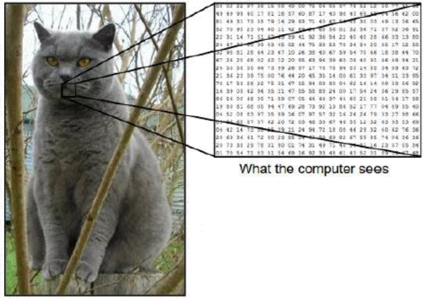


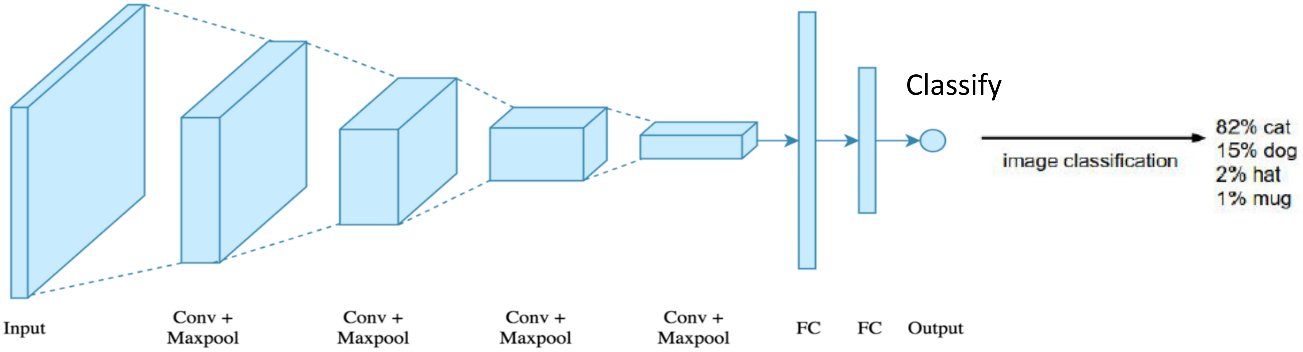
Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Convolutional Neural Network Architecture

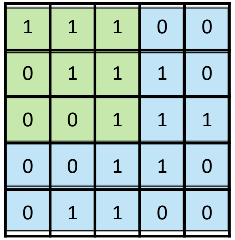


CNN Implementation

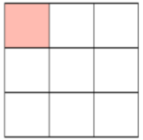




Convolution filters slides through the image



Stride 1

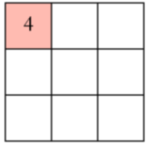


Feature Map

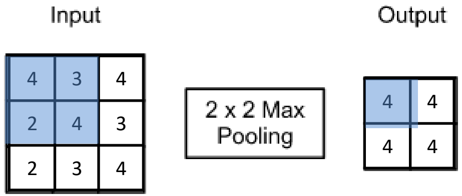
1	0	1
0	1	0
1	0	1

Filter

The convolution process

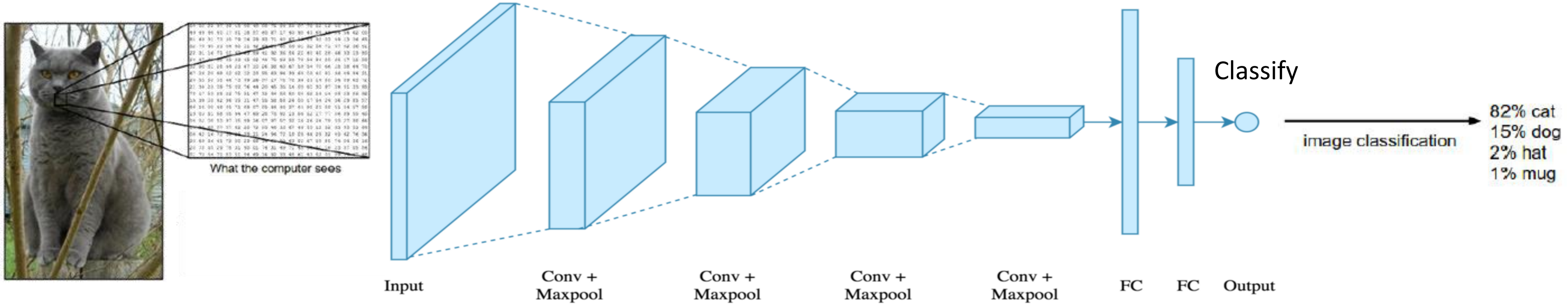


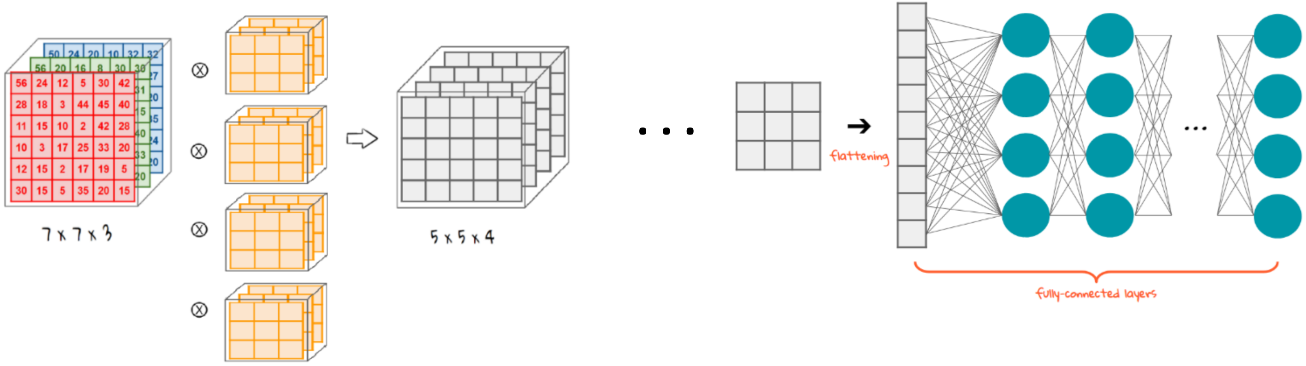
Feature map



The pooling process

CNN Implementation (Cont'd)





How Do Transformers Work

- Earlier forms of machine learning used a numerical table to represent each word
- This form of representation could not recognize relationships between words such as words with similar meanings
- This limitation was overcome by using multi-dimensional vectors, commonly referred to as word embeddings
- This method can represent words with similar contextual meanings or other relationships are close to each other in the vector space

How Do Transformers Work (Cont’d)

- Step 1: Converts text into word embeddings (numerical representations).
- Step 2: Encoder processes these embeddings, capturing context and relationships.
- Step 3: Decoder uses this context to generate unique, meaningful output.

Traditional vs LLM Word Embedding Techniques

Traditional Static Embeddings	Contextual Embeddings
Single vector representation for each word (static vectors)	Word vectors change based on context (dynamic vectors).
dog → [0.7, 0.2, 0.1, ...] cat → [0.8, 0.1, 0.05, ...] king → [0.5, 0.9, 0.3, ...] queen → [0.5, 0.8, 0.35, ...]	Sentence 1: "I went to the bank to deposit money." bank → [0.6, 0.2, 0.3, ...] Sentence 2: "The river bank was eroding." bank → [0.3, 0.5, 0.4, ...]

Word Embedding Process

Step 1: Pre-process Text as Numerical Representations (Using Word Embeddings)	**Text Input: "The quick brown fox jumps over the lazy dog."**	**Word Embedding: Each word is converted to a numerical vector representation.**	**"The quick brown fox jumps over the lazy dog."** **↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓** **[0.1, 0.2, ...] [0.3, 0.4, ...] [0.5, 0.6, ...] ... [0.9, 1.0, ...]**
Step 2: Encoder Processes Text and Understands Context (Transformers with Encoder)	**Word Embeddings:** Input word embeddings are processed by the encoder.	**Context Understanding**: The encoder captures relationships and context, such as similar meanings and parts of speech.	[0.1, 0.2, ...] [0.3, 0.4, ...] [0.5, 0.6, ...] ... ↓ ↓ ↓ Encoder (Self-Attn) Encoder (Self-Attn) Encoder (Self-Attn) ↓ ↓ ↓ [0.15, 0.25, ...] [0.35, 0.45, ...] [0.55, 0.65, ...] ...
Step 3: Decoder Produces Unique Output (LLMs with Decoder)	**Contextual Representations**: The decoder uses the context-aware representations from the encoder.	**Output Generation**: The decoder generates a unique output based on the encoded context and learned language patterns.	[0.15, 0.25, ...] [0.35, 0.45, ...] [0.55, 0.65, ...] ... ↓ ↓ ↓ Decoder (Self-Attn) Decoder (Self-Attn) Decoder (Self-Attn) ↓ ↓ ↓ "A unique sentence or output based on context."

How are LLMs Trained

- Transformer-based neural networks contain billions of parameters (weights and biases).
- Training is performed using a large corpus of high-quality data.
- During training, the model iteratively adjusts parameter values until the model correctly predicts the next token from the previous sequence of input tokens.
- It does this through self-learning techniques (back propagation) which teach the model to adjust parameters to maximize the likelihood of the next tokens in the training examples.

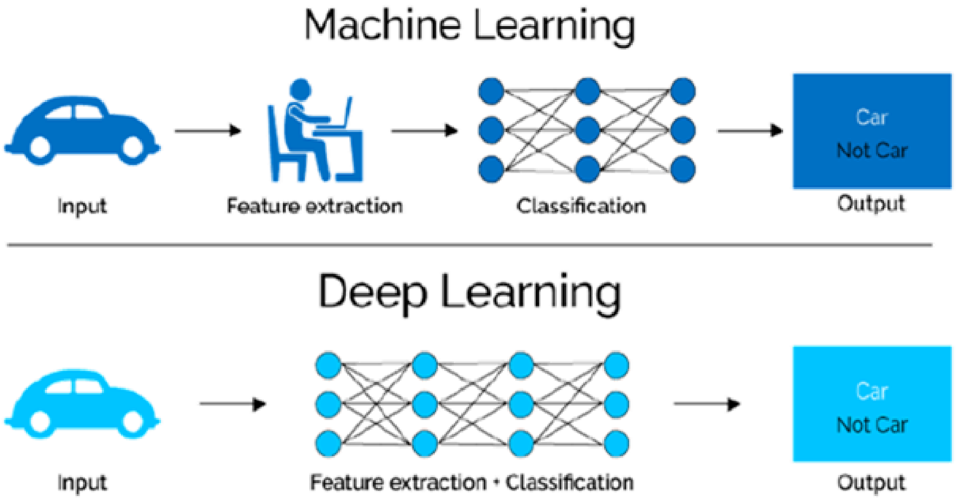
How are LLMs Trained (Cont’d)

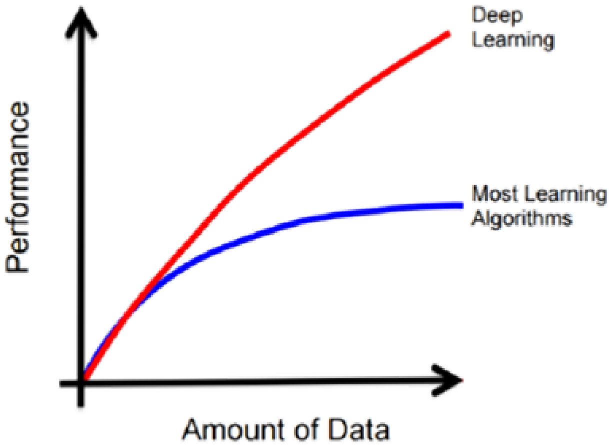
- Once trained, LLMs can be readily adapted to perform multiple tasks using relatively small sets of supervised data, a process known as fine tuning.
- Three common learning models exist:
  - Zero-shot learning: Base LLMs can respond to a broad range of requests without explicit training, often through prompts, although answer accuracy varies.
  - Few-shot learning: By providing a few relevant training examples of a new task at inference time, base model performance significantly improves in that specific area.
  - Fine-tuning: This is an extension of few-shot learning in that data scientists train a base model using specific dataset for a particular task to adjust its parameters with additional data relevant to the specific application.

Categories of LLMs

Model	Zero-shot Learning	Few-shot Learning	Fine-tuning
GPT-3 (OpenAI)	Yes	Yes	Yes
GPT-4 (OpenAI)	Yes	Yes	Yes
Claude (Anthropic)	Yes	Limited	Yes
PaLM 2 (Google)	Limited	Yes	Yes
BERT (Google)	Limited	Limited	Yes
T5 (Google)	Limited	Limited	Yes
LLaMA (Meta)	Limited	Limited	Yes
Gemini 1 (Google)	Limited	Limited	Yes
Mistral (Mistral)	Limited	Limited	Yes

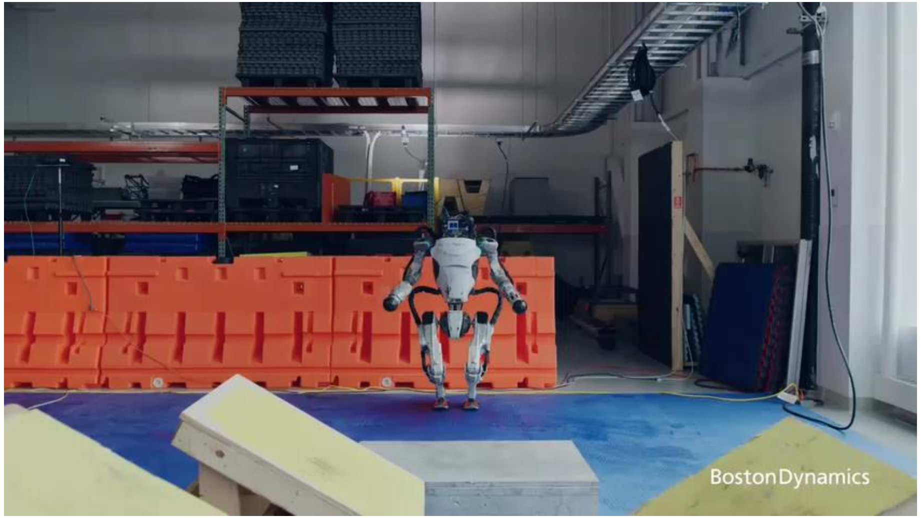
Why Deep Learning? Scalable ML



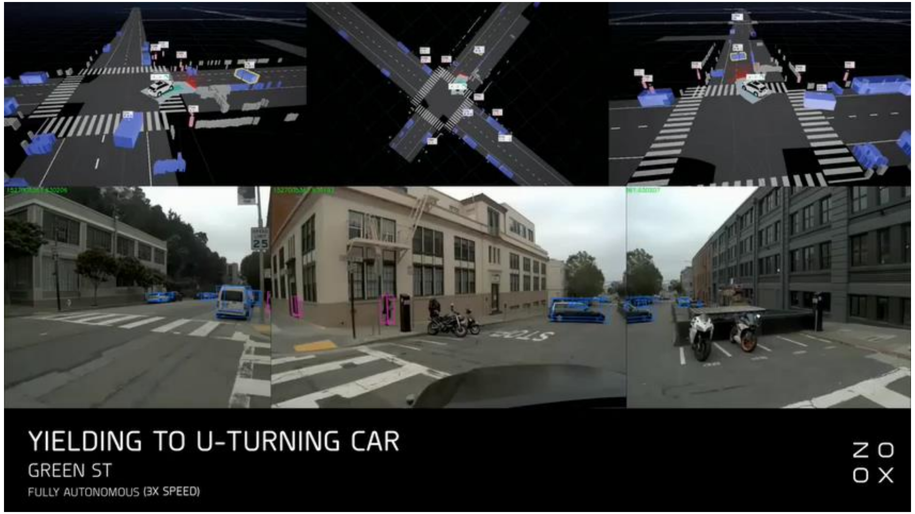


4 Applications of AI and ML

Why not Deep Learning? Real-World Applications



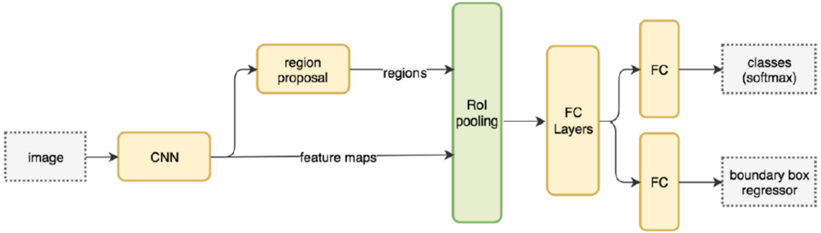
https://www.youtube.com/watch?v=tF4DML7FIWk



https://www.youtube.com/watch?v=868tExoVdQw

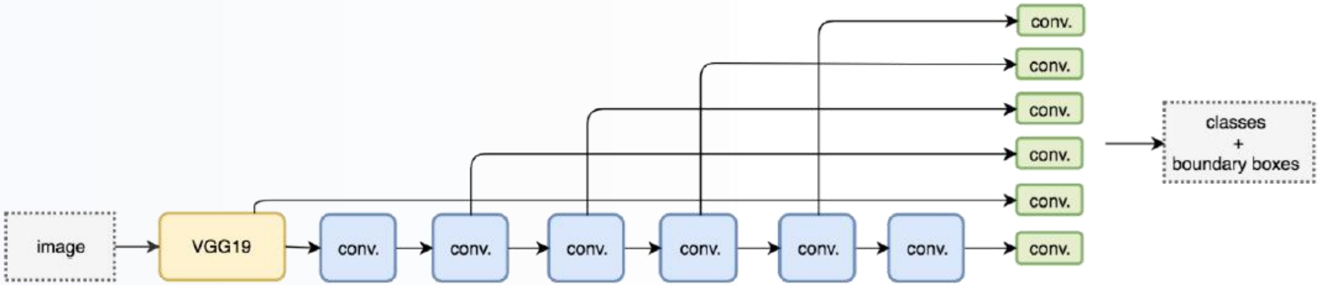
Image Detection / Localization with Region-Based Methods (Faster R-CNN)

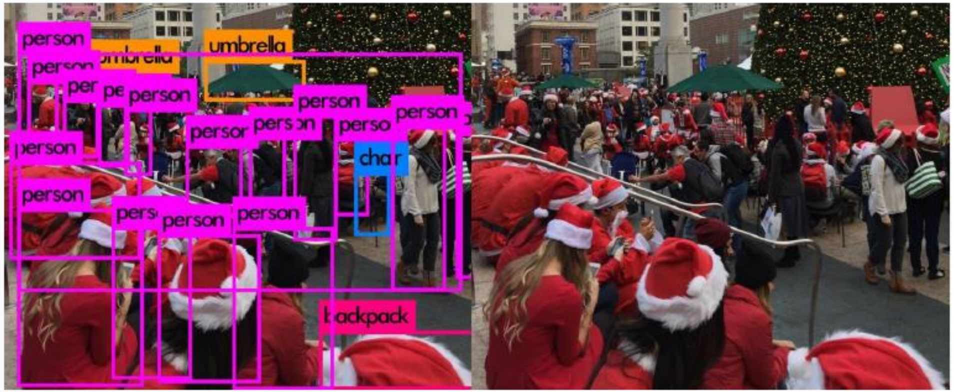
```
ROIs = region_proposal(image)
for ROI in ROIs
    patch = get_patch(image, ROI)
    results = detector(patch)
```



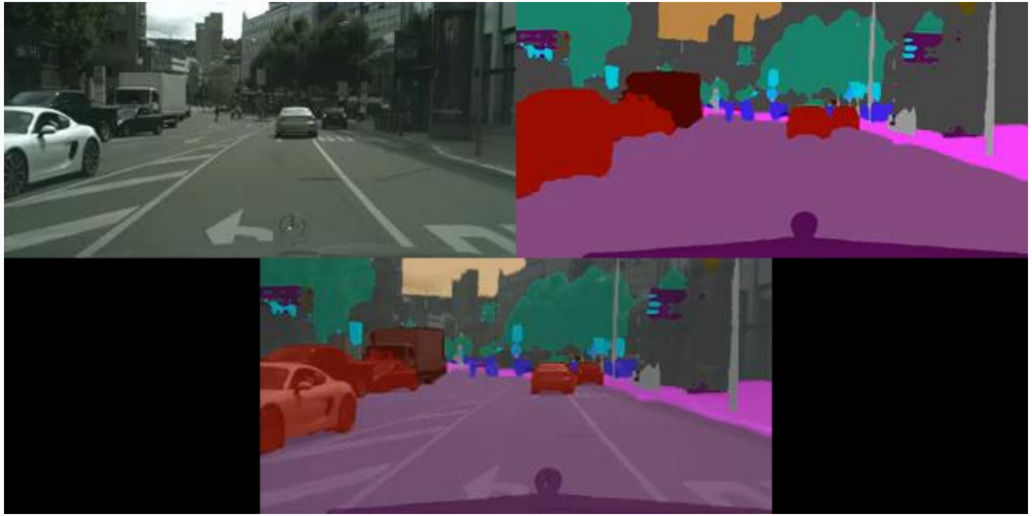
https://www.youtube.com/watch?v=zrHfvVdvmz4&t=1s

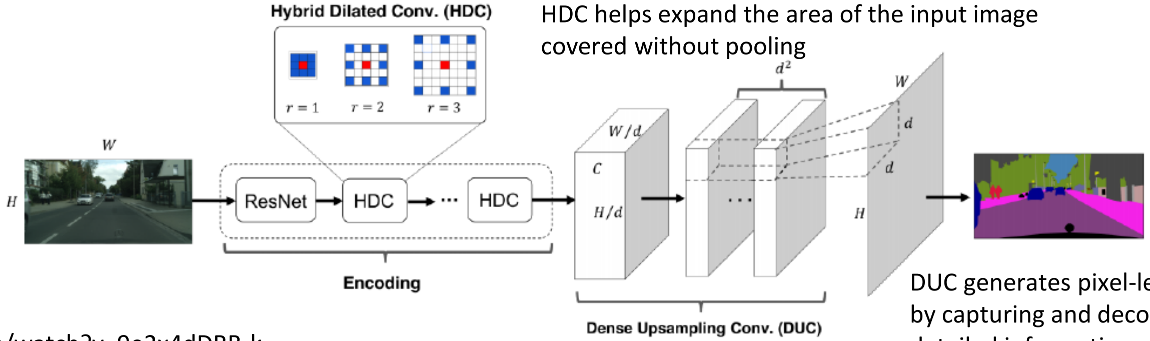
Image Detection / Localization with Single Shot Detection Methods (SSD)





Semantic Segmentation



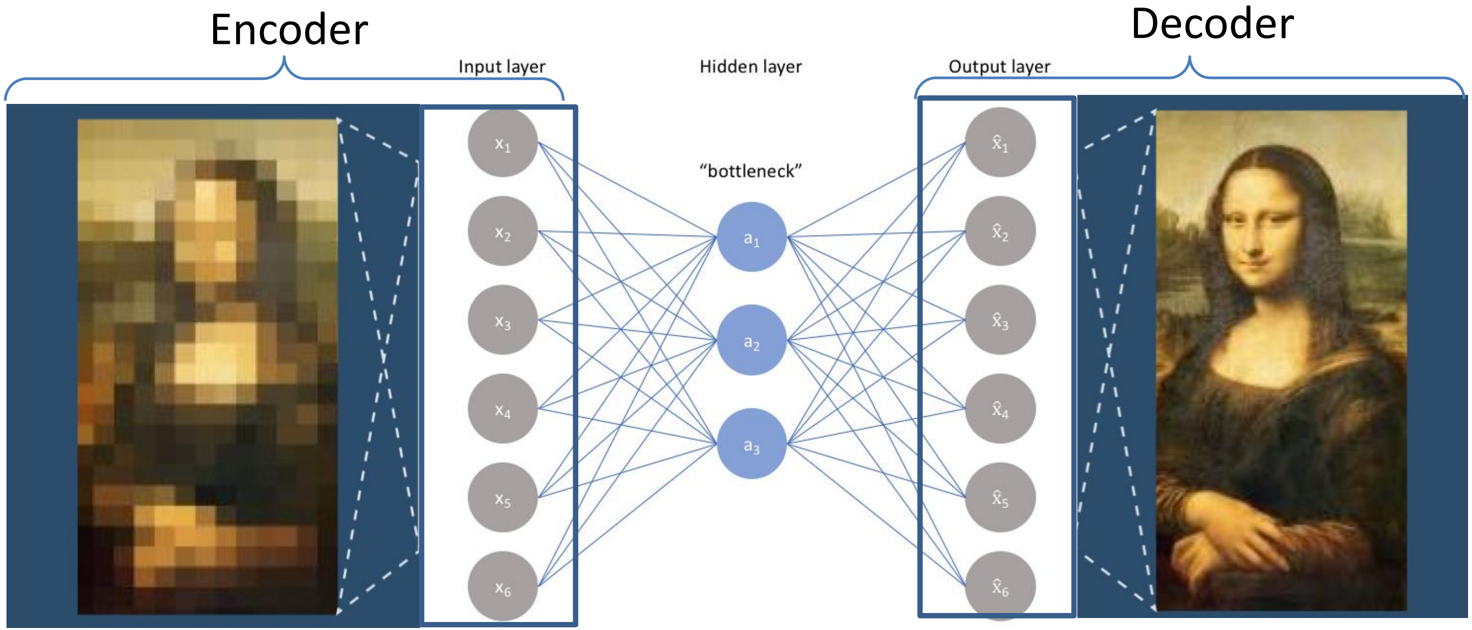


HDC helps expand the area of the input image covered without pooling

DUC generates pixel-level prediction by capturing and decoding more detailed information

https://www.youtube.com/watch?v=9e2x4dDRB-k

Autoencoders



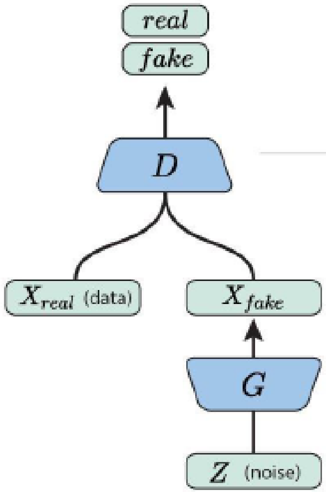
Encoder is a set of convolutional blocks followed by pooling modules that compress the input to the model

The ''bottleneck' restricts the flow of information to the decoder from the encoder, allowing only the most vital information to pass through.

Decoder upsampling the convolutional blocks and reconstructs the data back from its encoded form from the bottleneck's output.

Generative Adversarial Networks (GANs)

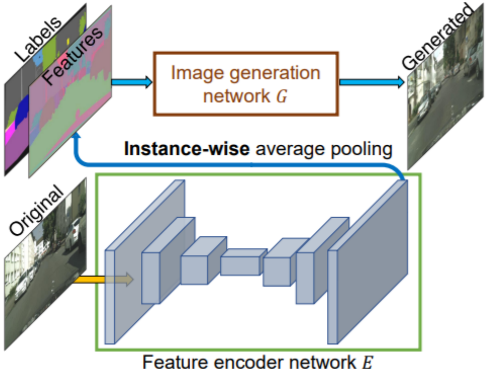
Generative Adversarial Networks (GANs) are a way to make a generative model by having two neural networks compete with each other.



The **discriminator** tries to distinguish genuine data from forgeries created by the generator.

The **generator** turns random noise into immitations of the data, in an attempt to fool the discriminator.

Generative Adversarial Networks (GANs) (Cont'd)



T-C Wang et al. "High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs" in CVPR, 2018.

Applications of Generative Adversarial Networks (GANs)

Generate Photographs of Human Faces

T. Karras, et al., "Progressive Growing of GANs for Improved Quality, Stability, and Variation", 2017.

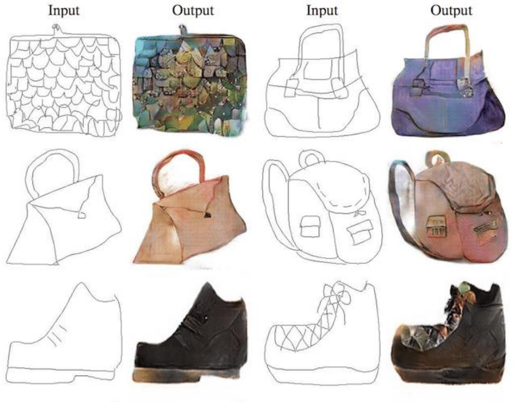
Generate Realistic Photographs

A. Brock, et al., "Large Scale GAN Training for High Fidelity Natural Image Synthesis", 2018.

Generate Consistent Video

https://www.youtube.com/watch?v=9reHvktowLY

Translation of sketches to color photographs



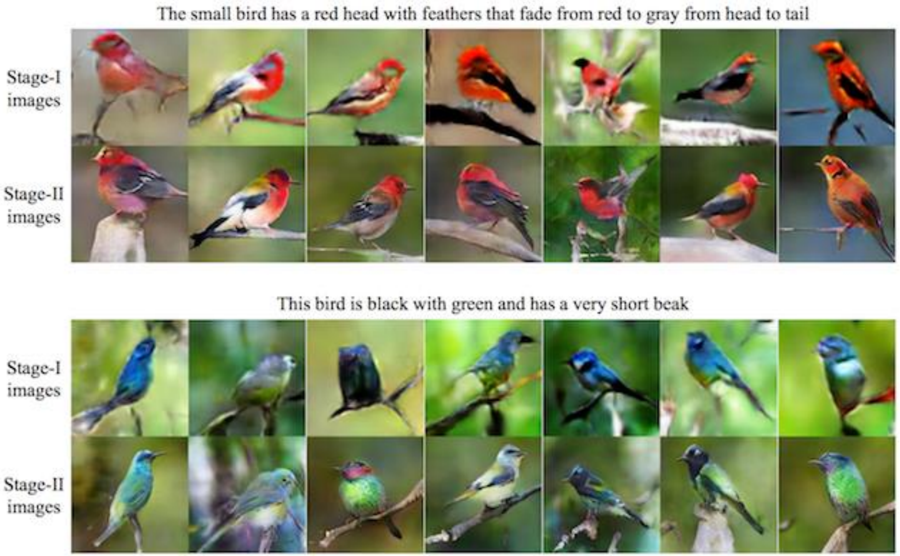
Phillip Isola, et al., "Image-to-Image Translation with Conditional Adversarial Networks", 2016.

Applications of Generative Adversarial Networks (GANs)

Generate Cartoon Characters

Y-H Jin, et al., “Towards the Automatic Anime Characters Creation with Generative Adversarial Networks” , 2017.

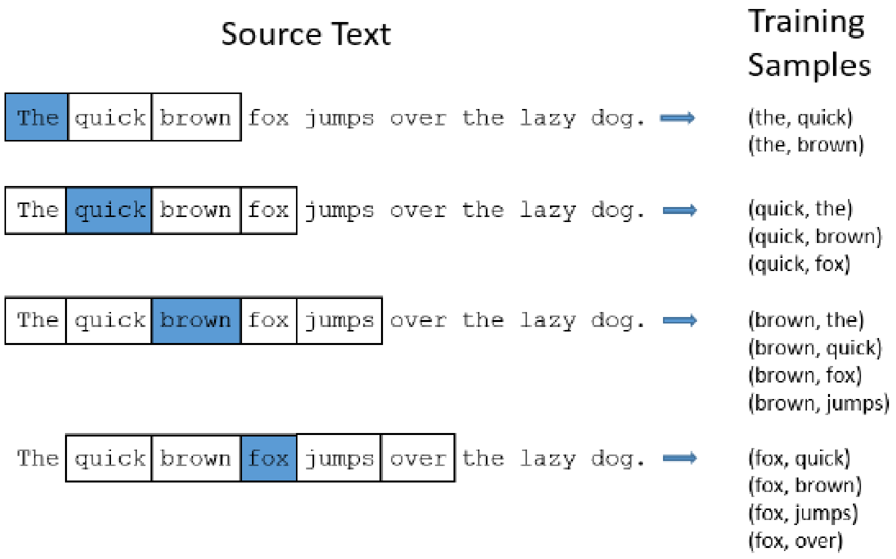
Text-to-Image Translation



Han Zhang, et al., “StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks”, 2016.

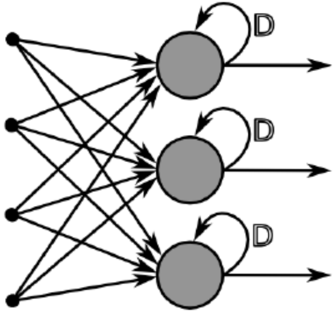
Word Embeddings (Word2Vec)

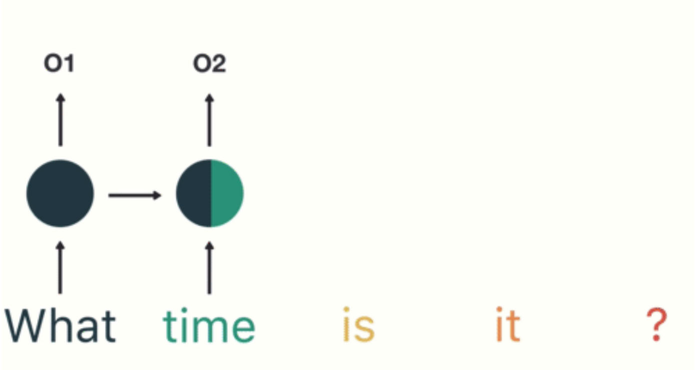
- Word embedding is capturing context of a word in a document, semantic and syntactic similarity, relation with other words
- Word2Vec is a method to construct an embedding. Another way to put it is, they are vector representations of a particular word.
- It is used to predict the source context words (surrounding words) given a target word (the center word).
- E.g. to predict the context [quick, fox] given target word 'brown' or [the, brown] given target word 'quick

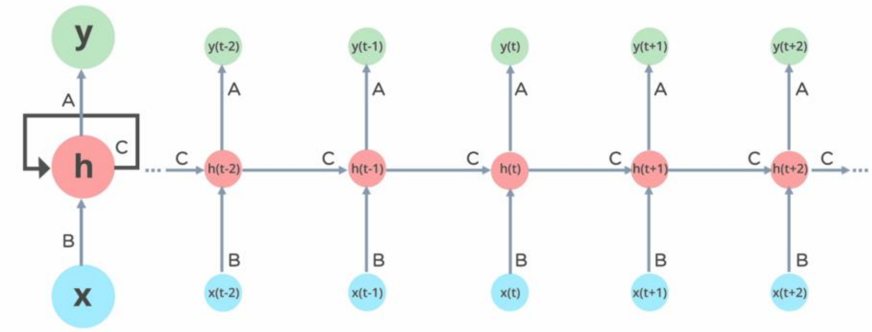


Skip Gram Model

Recurrent Neural Networks (RNNs)







Applications

- Sequence Data
- Text
- Speech
- Audio
- Video
- Generation

Long Term Dependency

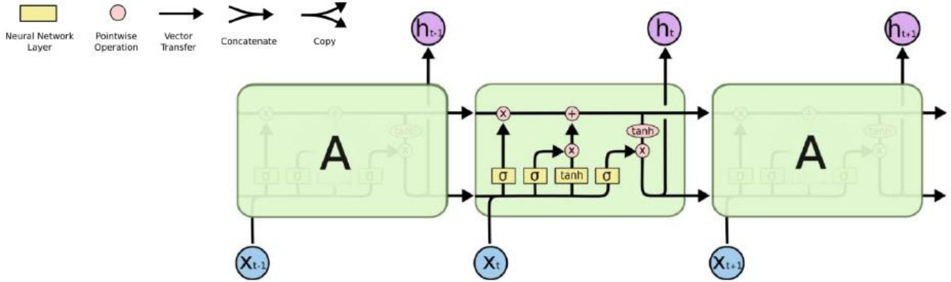
Short-term dependence:

Bob is eating an apple

Context

Long-term dependence:

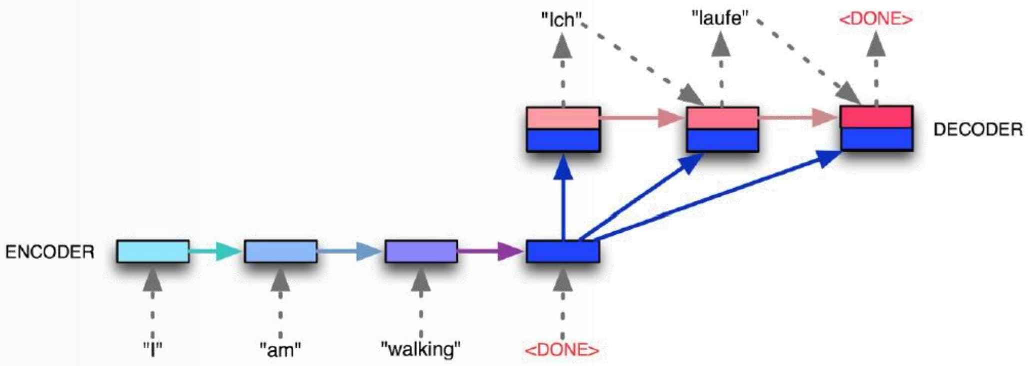
Bob likes apples. He is hungry and decided to have a snack. So now he is eating an apple.



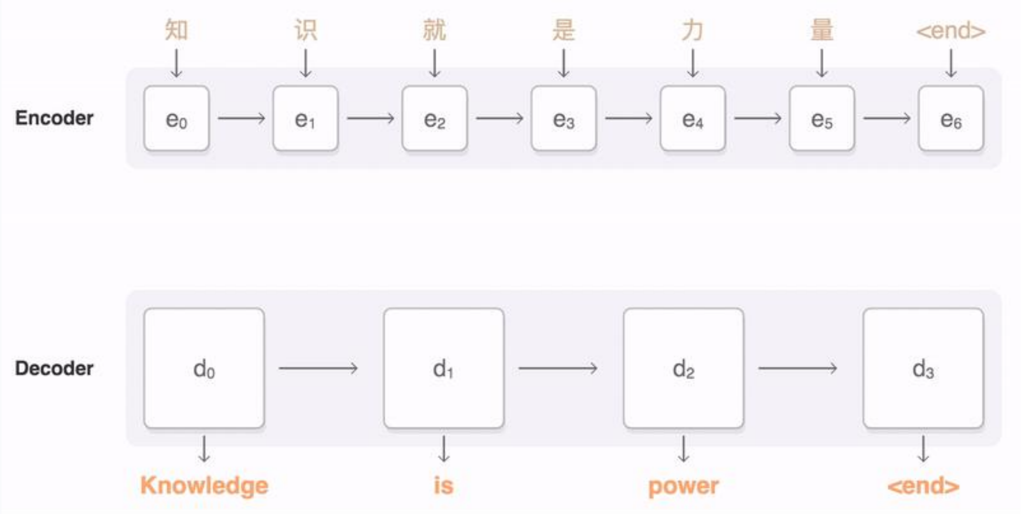
Long Short-Term Memory (LSTM) Networks:

Pick What to Forget and What To Remember

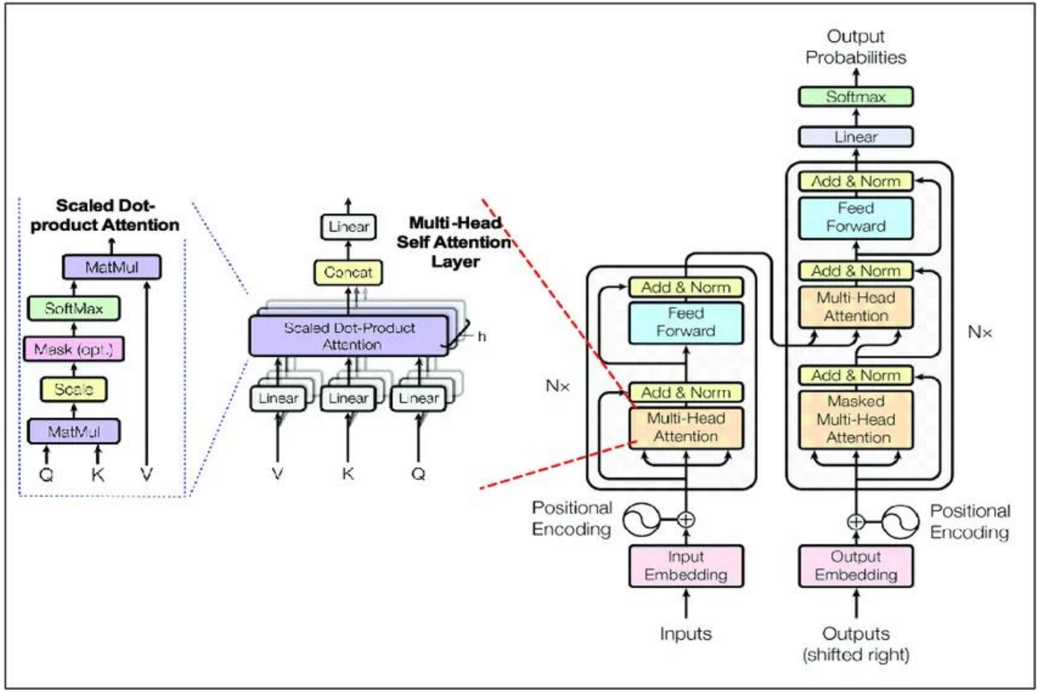
Encoder-Decoder Architecture



Machine Language Translation

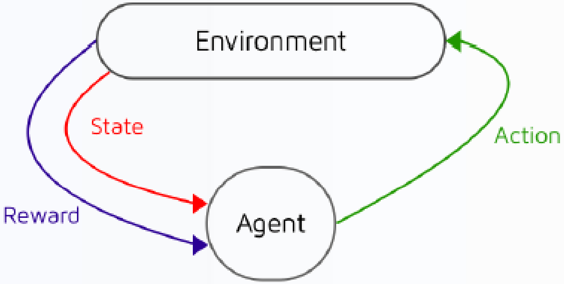


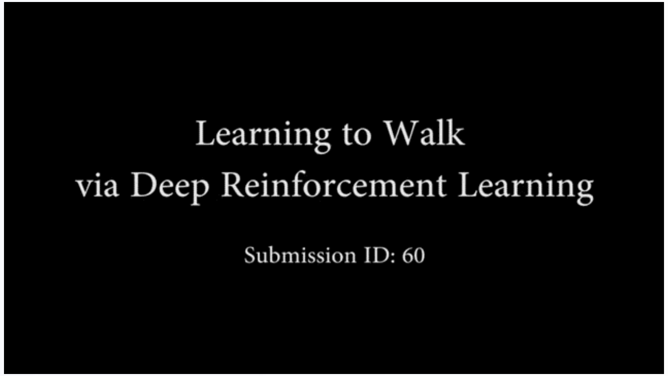
Large Language Models



- **Input Embeddings:** involves converting the input sentence into numerical embeddings, representing the semantic meaning of tokens within the sequence.
- **Positional Encoding:** To understand the sequential order of words, the input undergoes positional encoding.
- **Self-Attention:** allowing the model to weigh the significance of individual words in the input sequence focusing on relevant words and capture intricate relationships between them.
- **Feed-Forward Neural Networks:** utilizes neural networks to enhance the information contained in the representations
- **Output Layer:** The final output is generated based on the transformed representations obtained through the preceding steps, reflecting the model's interpretation of the input sentence.

Deep Reinforcement Learning

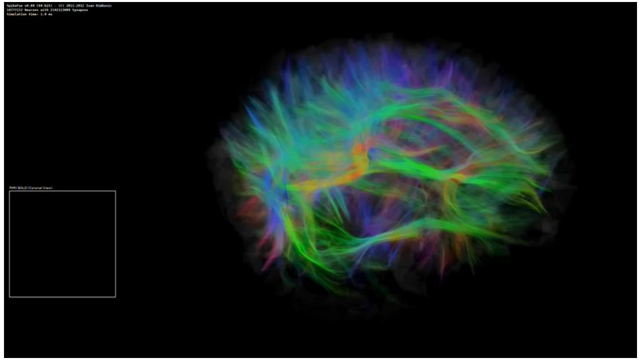


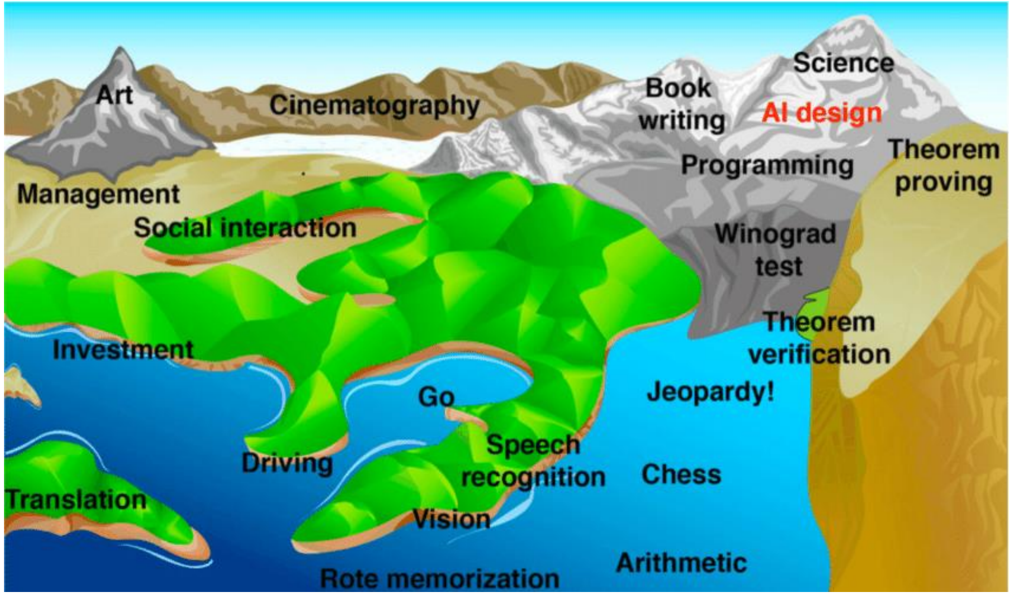


https://www.youtube.com/watch?v=n2gE7n11h1Y

https://www.facebook.com/watch/?v=344186809644234

Towards General AI





Hans Moravec's illustration of the rising tide of the AI capacity increasingly covering the landscape of human competence

Summary

- AI technology is already used in many fields, including medicine, the automotive industry, manufacturing, retail, social media, banking and finance sectors to mention just a few
- Machine learning depends on algorithms or a set of rules or sequence of instructions guiding an operation. The algorithm in ML is learned by searching for patterns in a training dataset that match the labels to the given input values.
- Deep learning is a is a subset of machine learning that uses artificial neural networks inspired by the human brain, to learn from large amounts of data to solve any pattern recognition problem without human intervention.

Quiz

- Deep learning is part of a broader family of ____________________ methods based on ____________________ with ____________________ that can be ____________________.

A. Machine Learning

B. Supervised or Unsupervised

C. Representation Learning

D. Artificial Neural Networks

Acronym References

- CNN Convolutional neural networks
- RBM Restricted Boltzmann machine
- RNN Recurrent neural network
- MNIST Modified National Institute of Standards and Technology
- LSTM Long short-term memory
- DBN Deep belief network
- GAN Generative adversarial network
- BERT Bidirectional Encoder Representations from Transformers
- CUDA Compute Unified Device Architecture
- CNTK Microsoft Cognitive Toolkit
- RCNN Region-based CNN
- HDC Hybrid Dilated Convolution
- DUC Dense Upscaling Convolution