Introduction to Deep Learning

C. Chandra Sekhar
Dept. of Computer Science and Engineering
Indian Institute of Technology Madras
Chennai-600036

chandra@cse.iitm.ac.in

Office Room: SSB 407

Regression and Classification Tasks

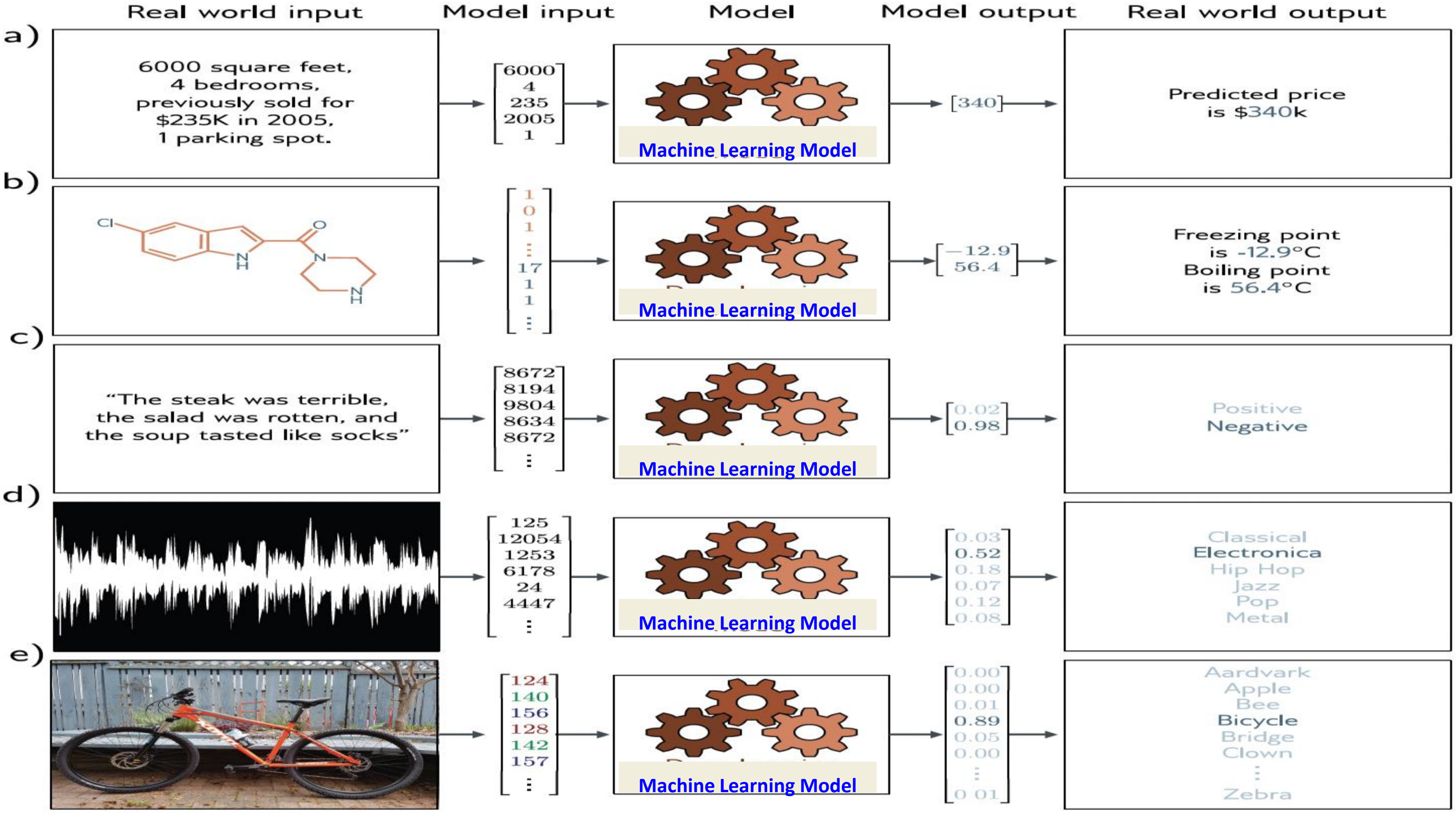


Figure 1.2 Regression and classification problems. a) This *regression* model takes a vector of numbers that characterize a property and predicts its price. b) This *multivariate regression* model takes the structure of a chemical molecule and predicts its melting and boiling points. c) This *binary classification* model takes a restaurant review and classifies it as either positive or negative. d) This *multiclass classification* problem assigns a snippet of audio to one of N genres. e) A second multiclass classification problem in which the model classifies an image according to which of N possible objects that it might contain.

S.J.D.Prince, Understanding Deep Learning, MIT Press, 2023

Learning Tasks with Structured Outputs

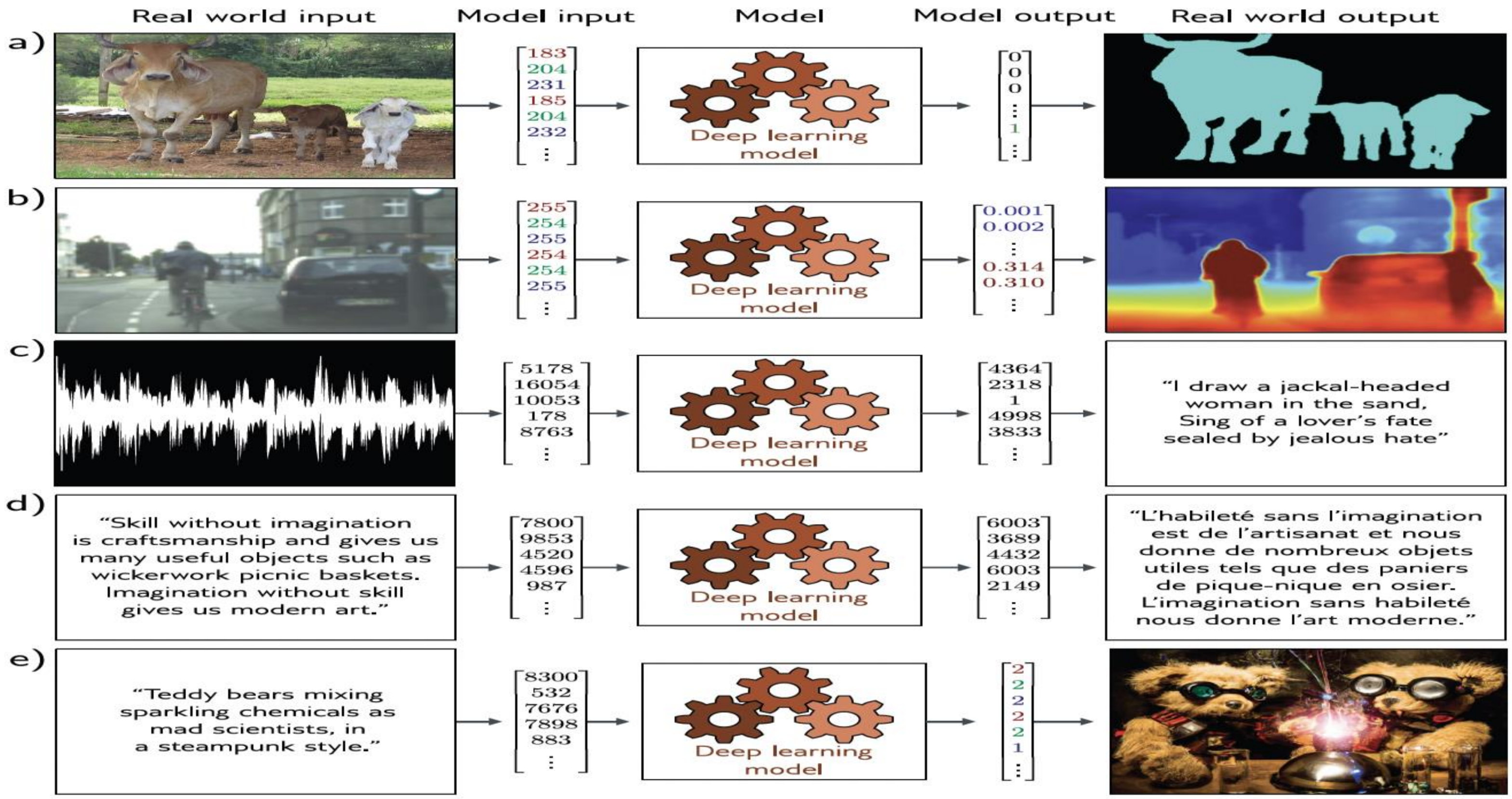


Figure 1.4 Supervised learning tasks with structured outputs. a) This semantic segmentation model maps an RGB image to a binary image indicating whether each pixel belongs to the background or a cow (adapted from Noh et al., 2015). b) This monocular depth estimation model maps an RGB image to an output image where each pixel represents the depth (adapted from Cordts et al., 2016). c) This audio transcription model maps an audio sample to a transcription of the spoken words in the audio. d) This translation model maps an English text string to its French translation. e) This image synthesis model maps a caption to an image (example from https://openai.com/dall-e-2/). In each case, the output has a complex internal structure or grammar. In some cases, many outputs are compatible with the input.

S.J.D.Prince, Understanding Deep Learning, MIT Press, 2023

Machine Learning Techniques for Classification

- **K-Nearest Neighbours Method**
- **Bayes Classifier**
  - **Statistical modeling**
  - **Unimodal distribution modeling**
  - **Multimodal distribution modeling: Gaussian Mixture Model**
- **Multilayer feedforward neural network based classification**
- **Support vector machine based classification**
- **Classification using decision tree**
  - **Random forest based classification**
- **Classification of sequential or temporal patterns**
  - **Hidden Markov model**

Image Classification

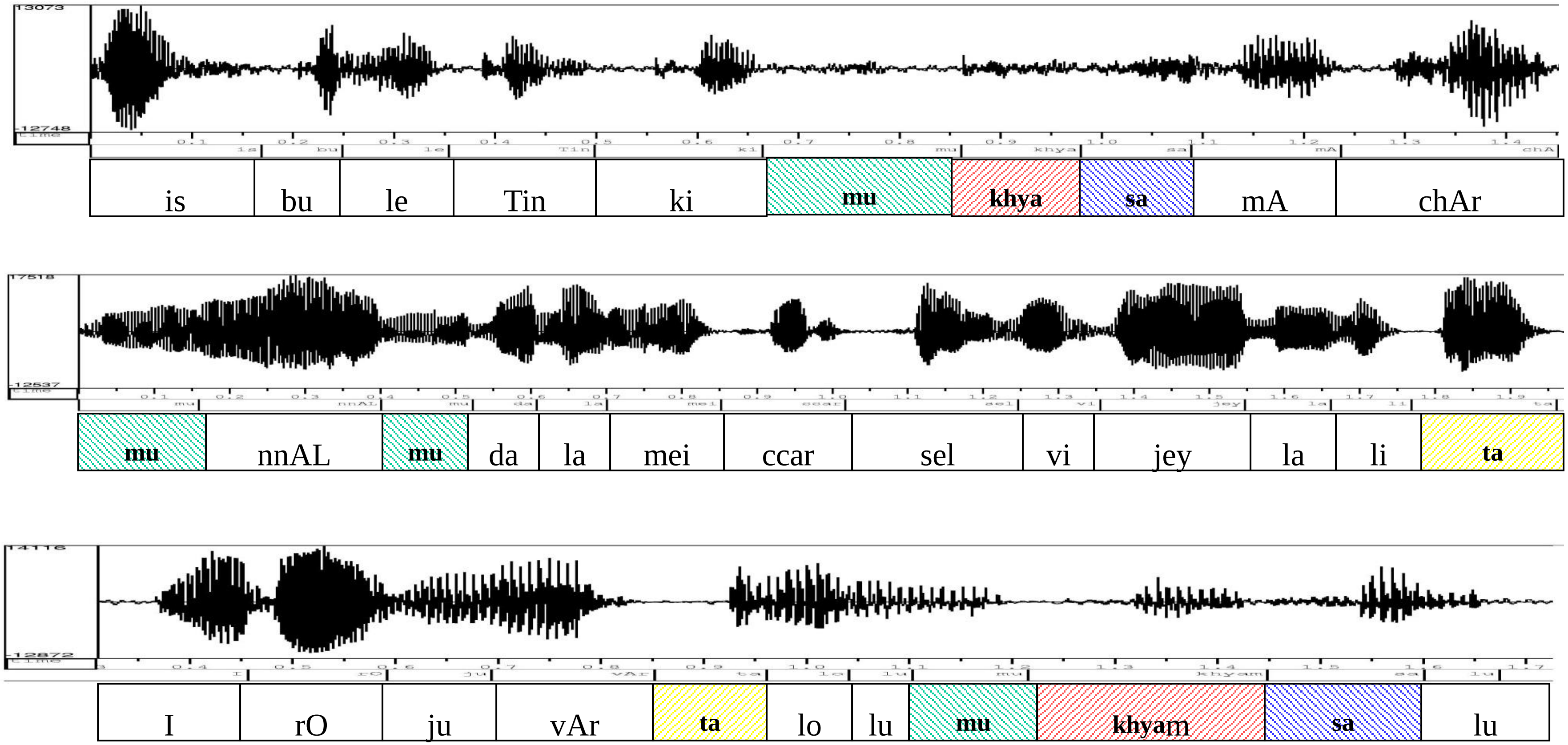
Tiger

Giraffe

Horse

Bear

Pattern Classification Tasks in Speech Processing



- Speech Recognition
- Speech Emotion Recognition
- Speaker Recognition
- Spoken Language Identification

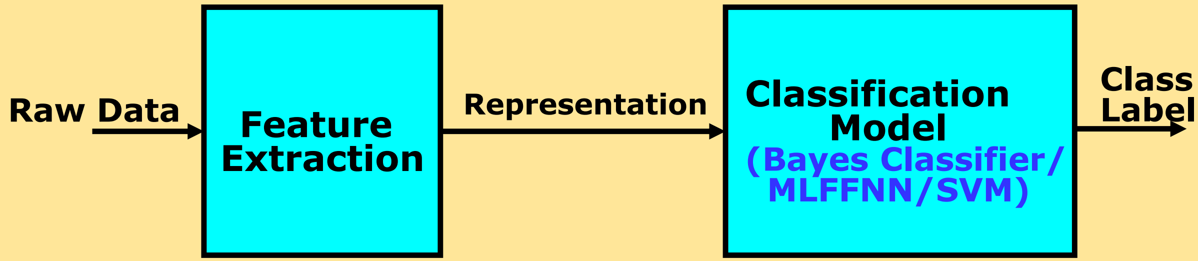
Text Processing Tasks

- **Sentence classification**
- **Parts-of-speech tagging**
- **Named entity recognition**
- **Sentiment analysis**

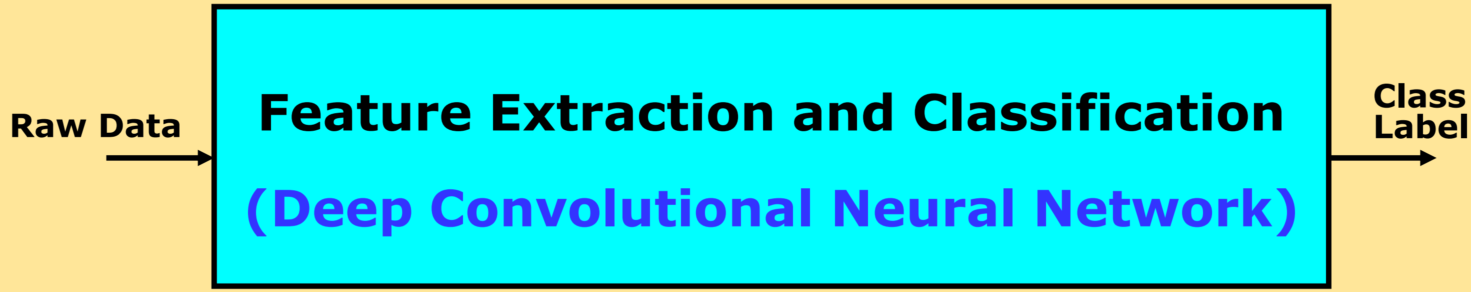
Classification using Deep Learning Models

Representation Learning: Conventional machine learning techniques (Bayes Classifiers, MLFFNNs and SVMs) take hand-designed features as input to models. Focus of deep learning techniques is to learn representation (features) from raw data given as input to models.

Conventional Approaches to Pattern Classification:

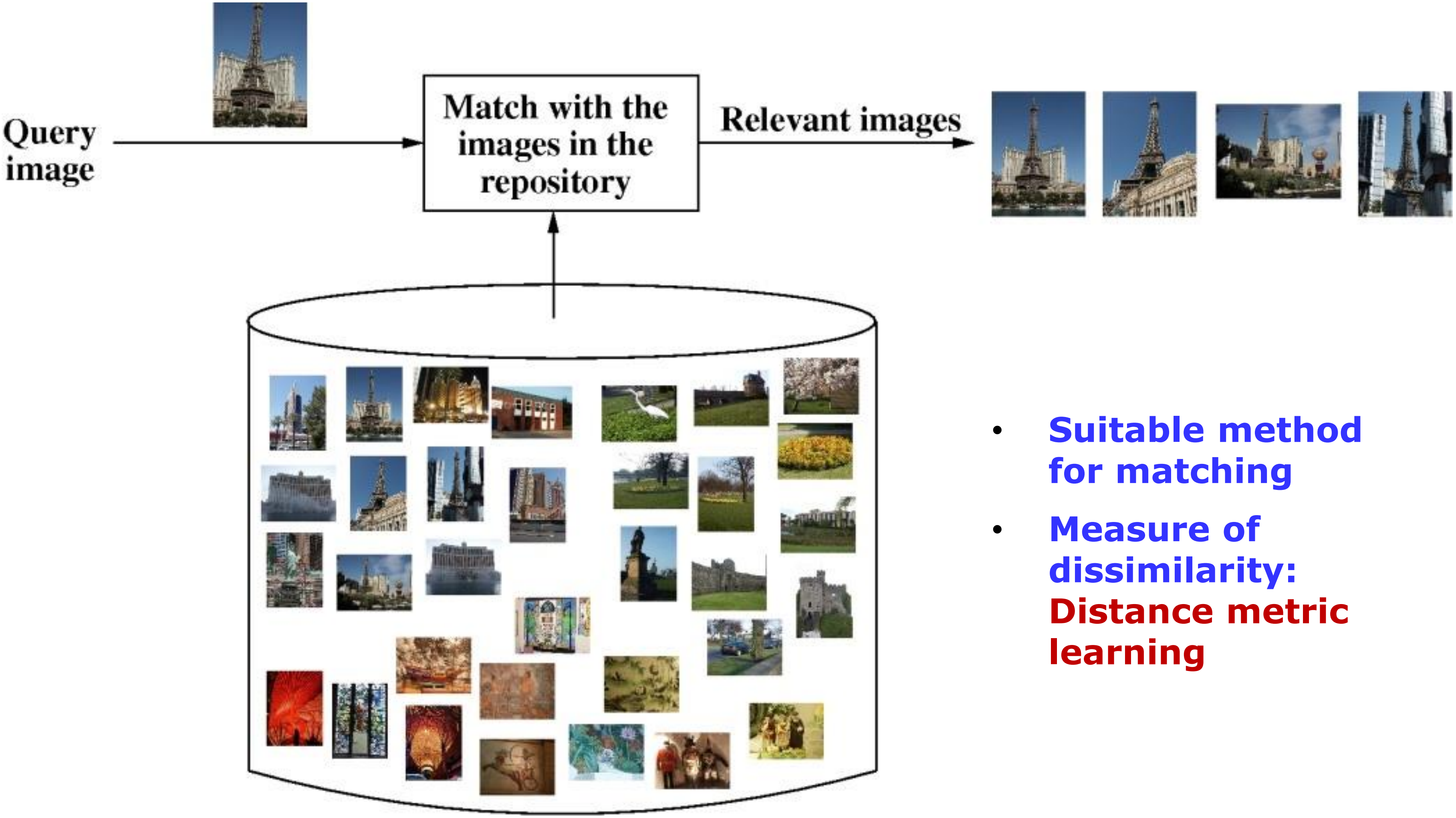


Deep Learning based Approaches to Pattern Classification:



Content based Image Retrieval

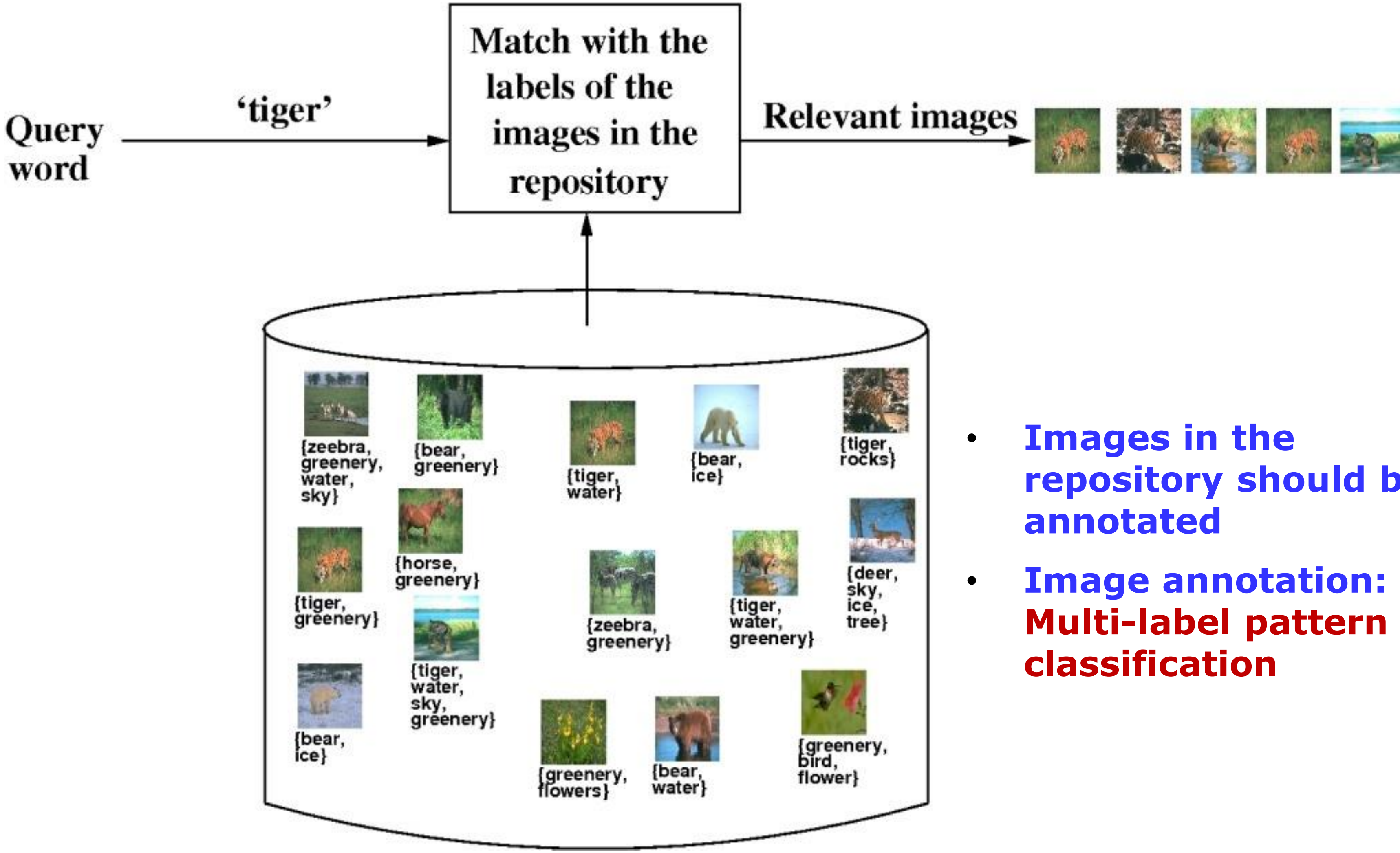
- **Query-by-example (QBE) Approach**



- **Suitable method for matching**
- **Measure of dissimilarity: Distance metric learning**

Content based Image Retrieval

- **Query-by-semantics (QBS) Approach**



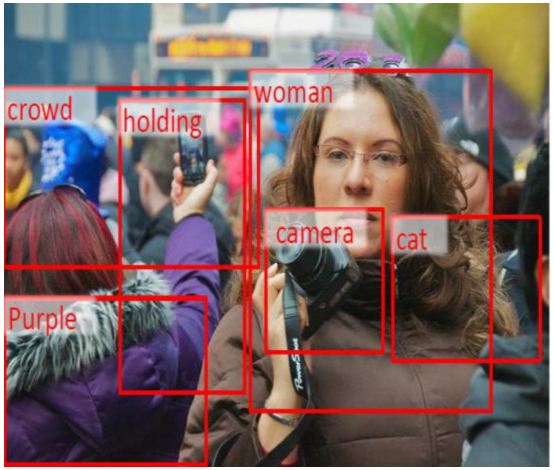
- **Images in the repository should be annotated**
- **Image annotation: Multi-label pattern classification**

Image Captioning

A group of people shopping at an outdoor market.

There are many vegetables at the fruit stand

O. Vinyals, A. Toshev, S. Bengio and D.Erhan, "Show and tell: Lessons learned from the 2015 MSCOCO Image Captioning Challenge," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no.4, pp.652-663, April 2017.

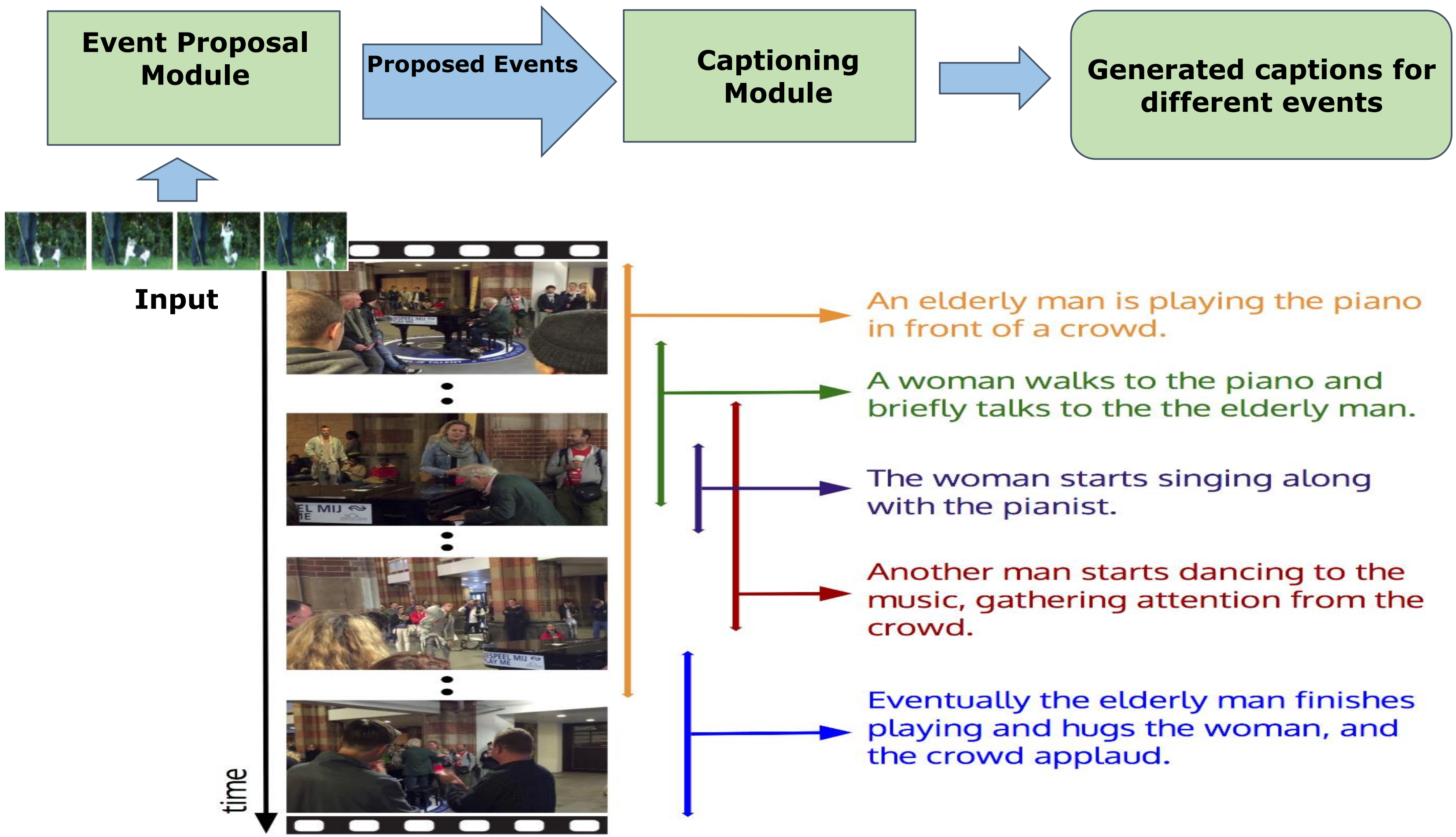


A woman holding a camera in crowd

Fang et al., "From captions to visual concepts and back", CVPR, 2015.

Video Captioning

- Generate text descriptions by localizing interesting events in a video.
  - **Event detection:** Event Proposal Module
  - **Event description:** Captioning Module



Visual Question Answering

Is there something to cut the vegetables with?

Yes

No

Who is wearing glasses?

Man

Woman

How many children are in the bed?

Two

One

Visual Commonsense Reasoning

1. Where is this happening?

a) There's a conference in this room.	**27.2%**
b) This is happening in a fancy restaurant.	**18.8%**
c) This is a wedding.	**52.8%**
d) This is happening in an industrial zone.	**1.3%**

I think so because...

a) You can see they are in a restaurant by the other tables, and you can tell it is a fancy restaurant by the wine in a bucket on his table. [person1] is obviously happy and is focused across his table.	**8.9%**
b) This is a formal setting and everyone is dressed nicely.	**1.6%**
c) [person1] is dressed fancily and the background is fancy.	**88.7%**
d) Drinking while in a car is illegal, and some restaurants have strange seating to draw in customers.	**0.8%**

Deep Learning Models

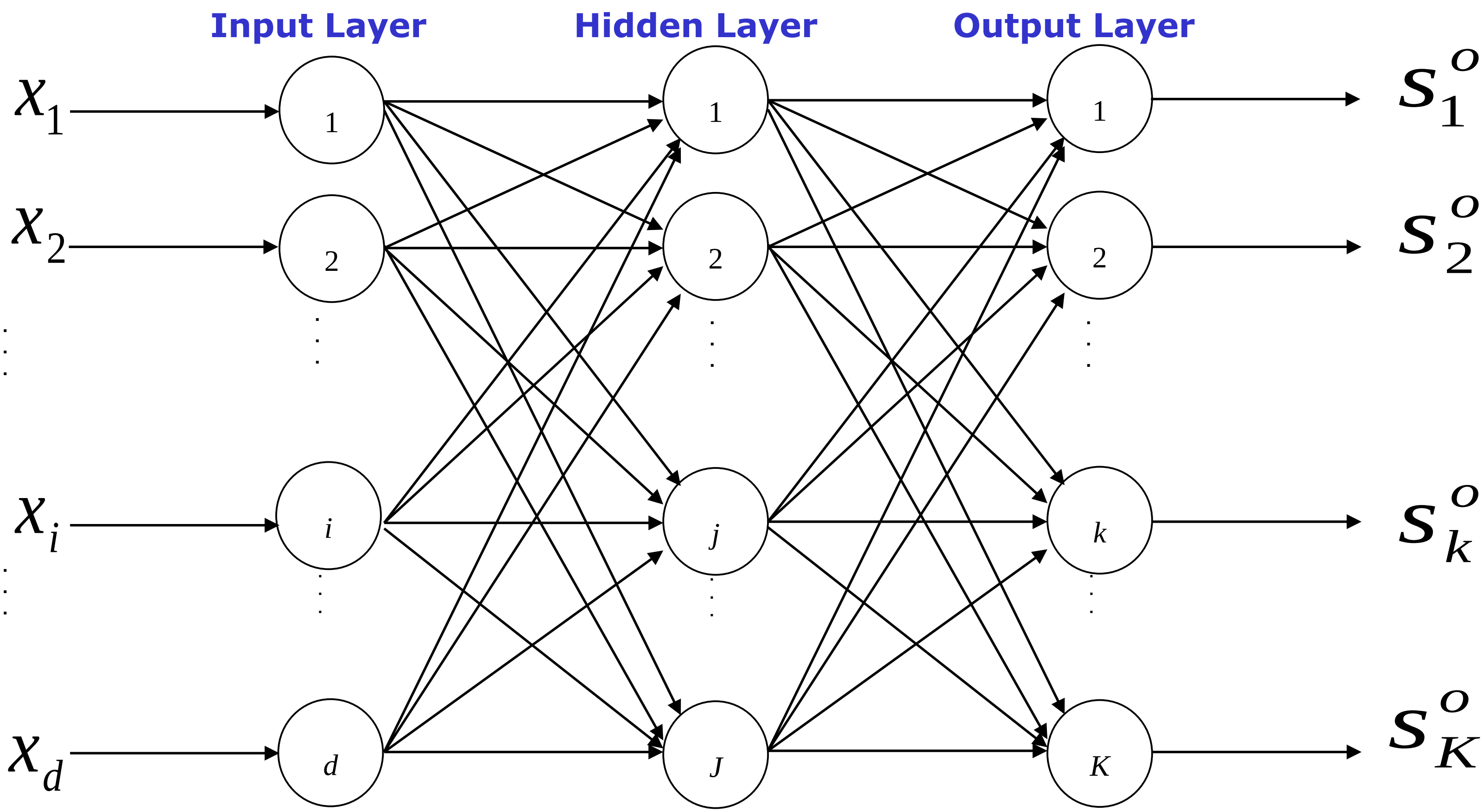
- Deep Feedfoward Neural Networks (DFNNs)
- Stacked Autoencoder based Pre-training for DFNNs
- Convolutional Neural Networks (CNNs)
- Recurrent Neural Networks (RNNs)
    - Long Short Term Memory (LSTM) Networks
- Attention based Models: Transformers
    - Pre-training of transformer model: BERT
- Generative Models
    - Generative Pre-trained Transformers (GPT)
    - Variational Autoencoders
    - Generative Adversarial Networks (GANs)
    - Diffusion Models

Learning Paradigms

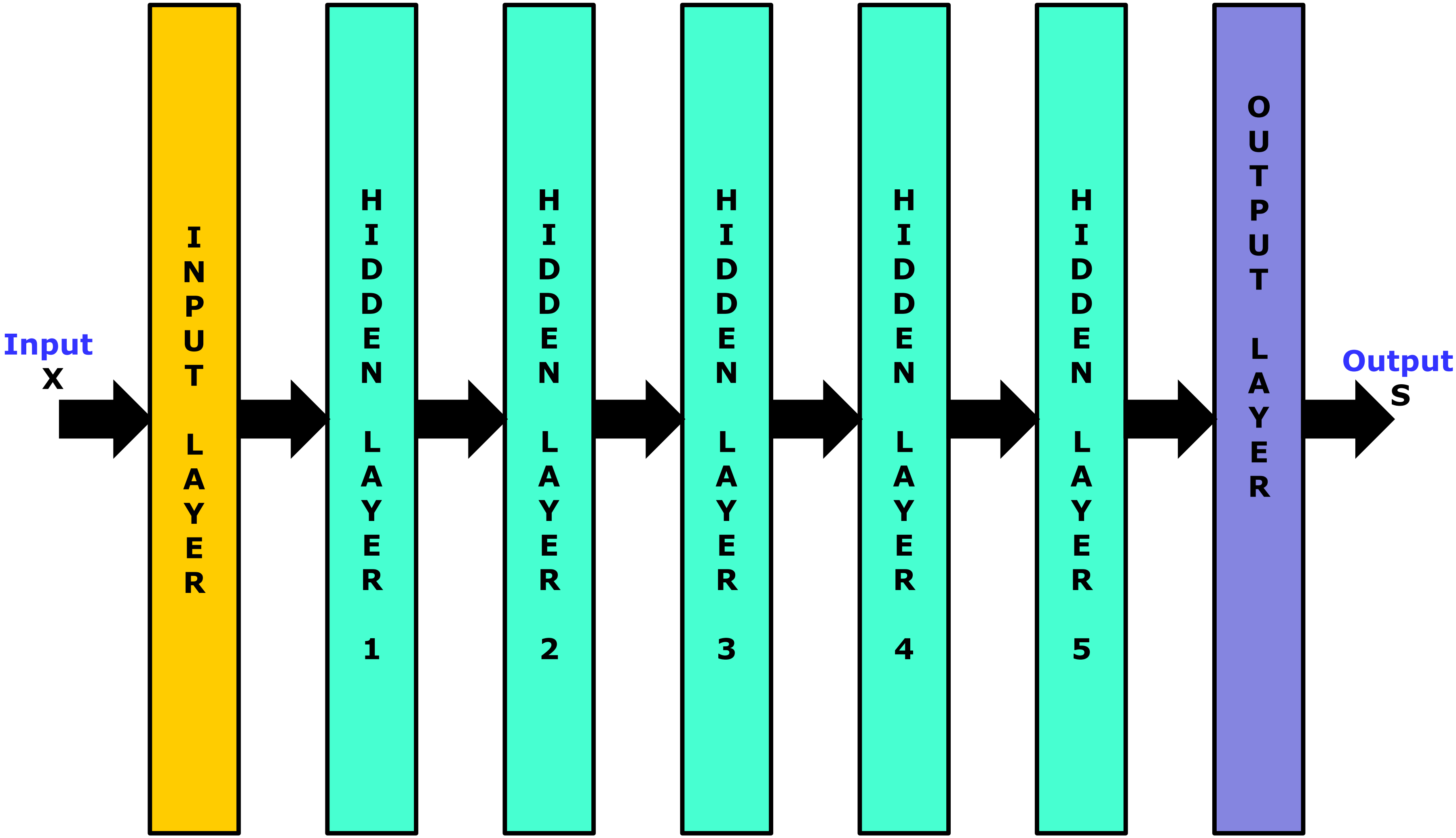
- **Learning Paradigms**
  - **Supervised learning**
  - **Unsupervised learning**
  - **Semi-supervised learning**
  - **Self-supervised learning**
  - **Adversarial learning**
  - **Transfer learning**
  - **Meta-learning**
  - **Active learning**
  - **Few-shot learning**
  - **Zero-shot learning**
  - **Federated learning**

Multilayer Feedforward Neural Network

- **Architecture of an MLFFNN**
  - **Input layer**: Linear neurons
  - **Hidden layers (1 or 2):** Sigmoidal neurons
  - **Output layer:** Sigmoidal neurons or Softmax neurons



Deep Feedforward Neural Network (DFNN)



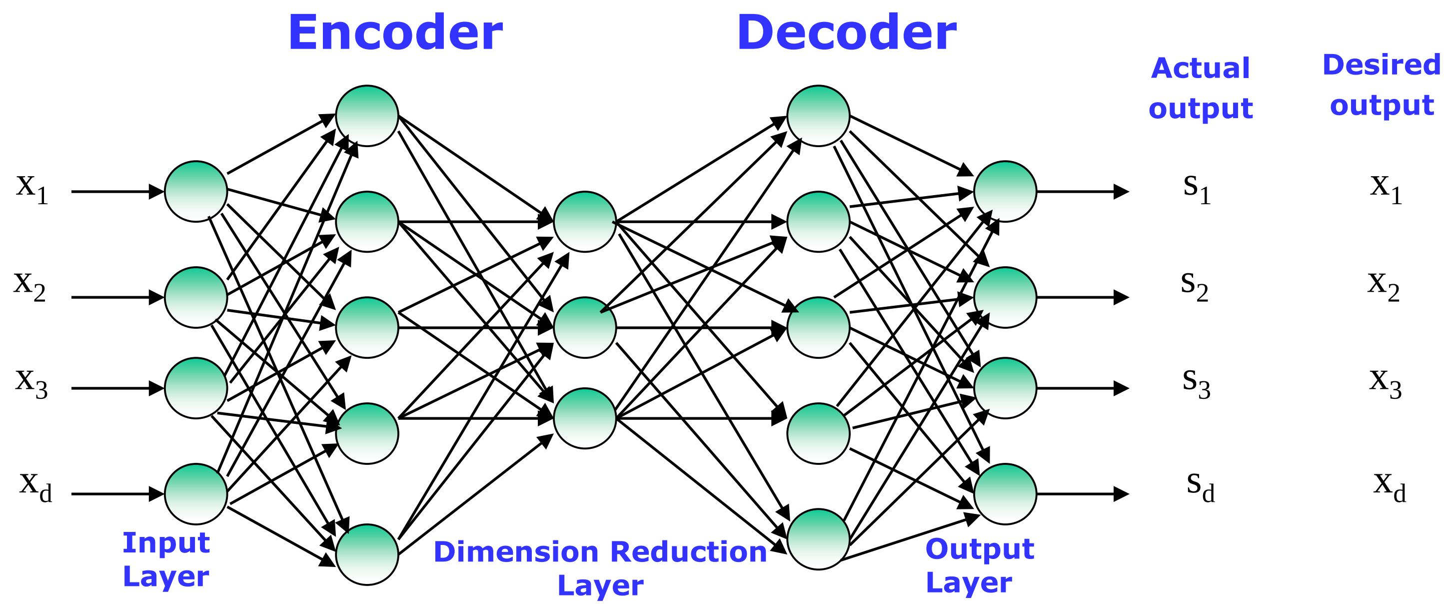
Optimization Methods for Training a DFNN

- Slow convergence of gradient descent method
- Problem addressed: How to reduce the number of epochs taken to reach a local minimum?
- Weight update methods that use the past history of updates have been shown to be effective.
- Generalized delta rule that uses momentum factor
- Weight-specific learning rate scheduling methods (Adaptive learning rate methods)
  - AdaGrad
  - RMSProp
  - AdaDelta
  - AdaM
- Second-order methods for optimization

Regularization Methods for Training a DFNN

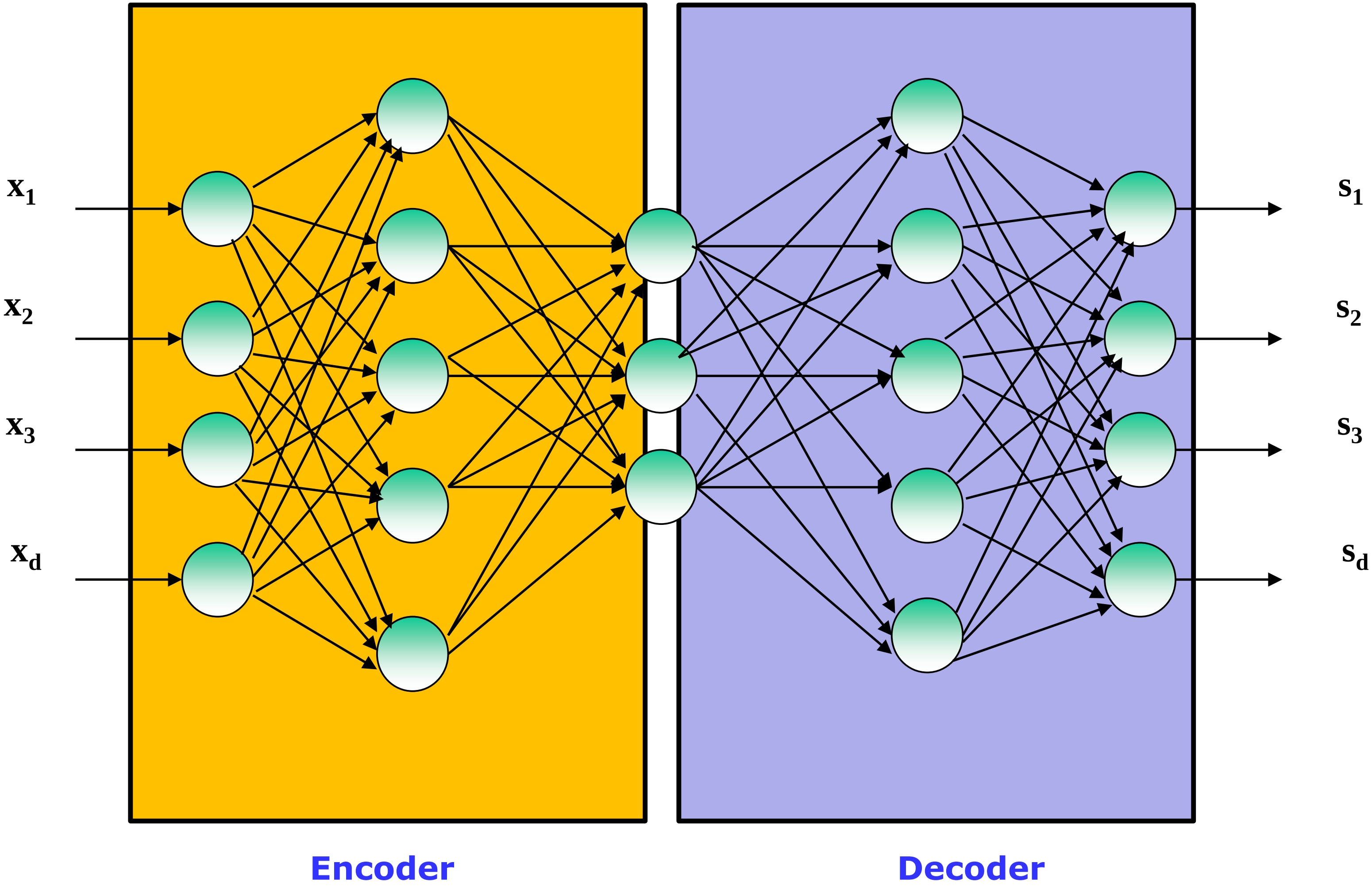
- **Underfitting: Model complexity is low**
- **Overfitting:**
  - **Model complexity is high**
  - **Training dataset size is small**
- **L2 regularization method**
- **Dropout method**
- **Drop connect method**
- **Batch normalization**

Auto-Association Neural Network (AANN)

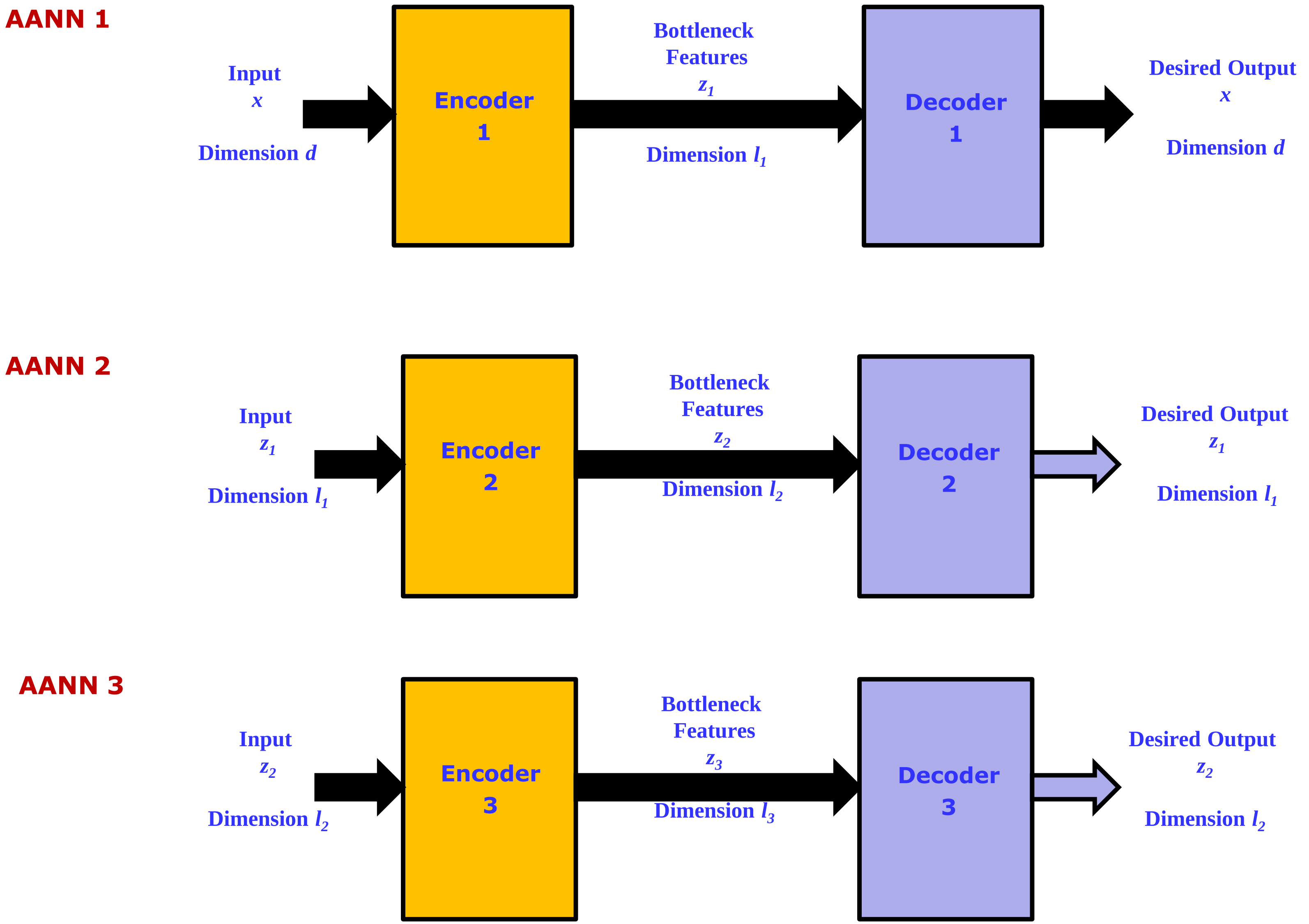


- **AANN uses linear neurons in the Input layer, Dimension reduction layer and Output layer. It uses sigmoidal neurons in the other two hidden layers.**
- **AANN is trained using the backpropagation learning method**
- **After the model is trained, the output of the Bottleneck Layer (Dimension reduction layer) is used as the reduced dimension representation of the input**
- **Encoder in AANN, also called as autoencoder, is used in Deep stacked autoencoder network models**

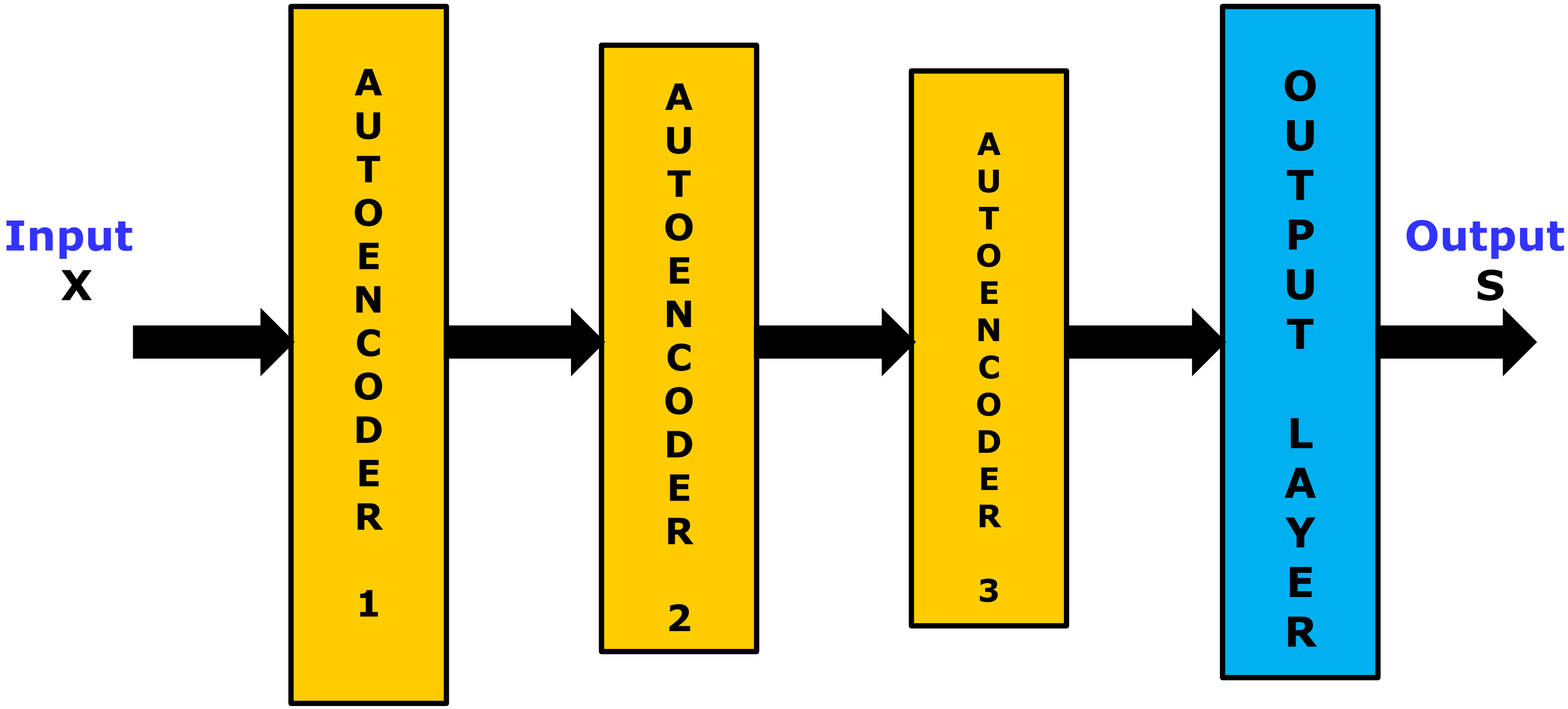
Auto-Association Neural Network (AANN)



Multiple AANNs for Stacked Autoencoder



Stacked Autoencoder for Pre-training a DFNN



- Weights of autoencoders are learnt using **unsupervised learning** with **unlabeled examples**. These weights are used as the **initial weights** for DNN.

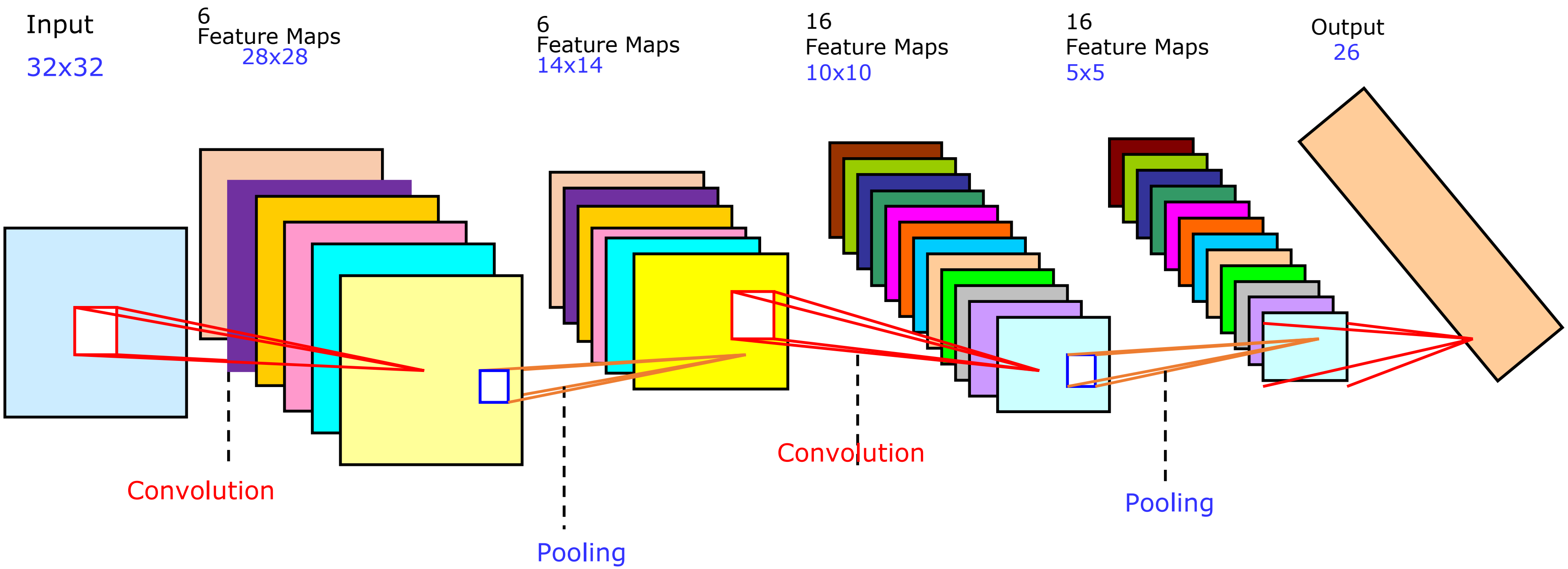
- **Fine-tuning** of DNN involves modification of weights using **backpropagation learning** method that uses a small set of **labeled examples**.

Convolution Neural Networks (CNNs)

- **Convolutional neural network (CNN) is a special type of multilayer feedforward neural network (MLFFNN) that is well suited for image classification.**
- **Development of CNN is neuro-biologically motivated.**
- **A CNN is an MLFFNN designed specifically to recognize 2-dimensional shapes with a high degree of invariance to translation, scaling, skewing and other forms of distortion.**

S. Haykin, *Neural Networks and Learning Machines*, Prentice-Hall of India, 2011

LeNet5: CNN for Handwritten Character Recognition



- Input: 32x32 pixel image of a character centered and normalized in size
- Weight sharing: All the nodes in a feature map in a convolutional layer have the same synaptic weights **(~278000 connections, but only ~1700 weight parameters**)
- Output layer: 26 nodes with one node for each character. Each node in the output layer is connected to the nodes in all the feature maps in the 4th hidden layer.

Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of IEEE, vol.86, no.11, pp.2278-2324, November 1998.

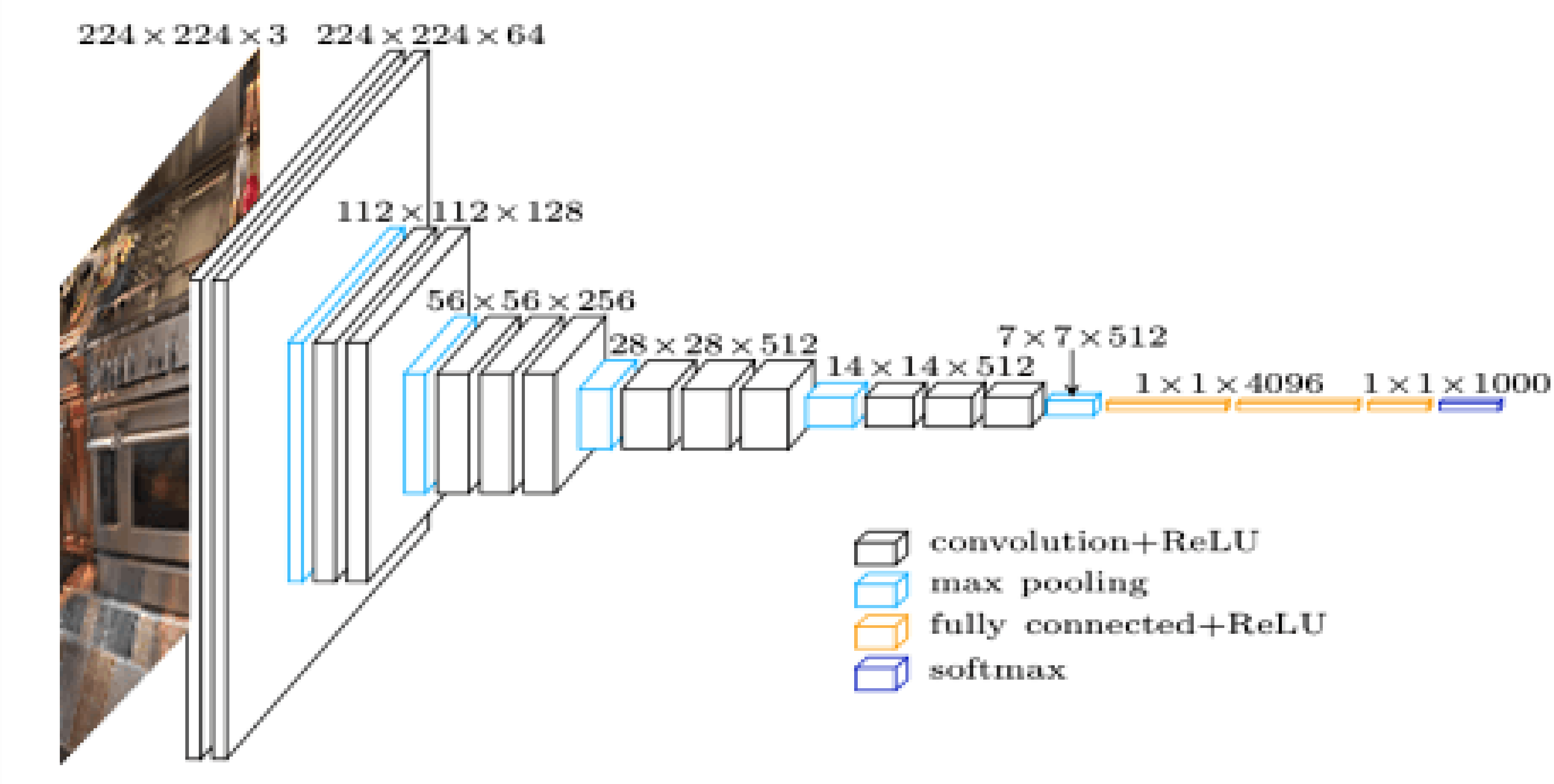
CNN Models for Image Classification

- **Image Classification (on ImageNet data):**
  - **AlexNet**
  - **VGG-Net**
  - **ResNet**
  - **GoogLeNet**
  - **PReLU-Net**
  - **Batch Normalization(BN)-Inception-ResNet**

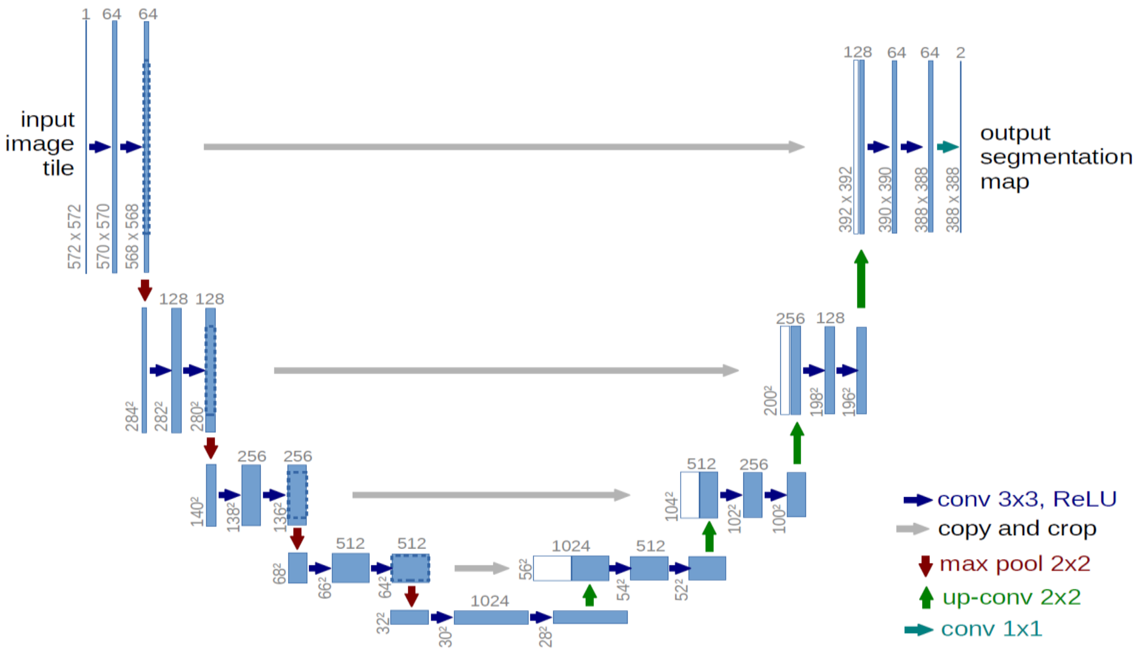
- **W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive survey," Neural Computation, vol.29, pp.2352-2449, 2017.**

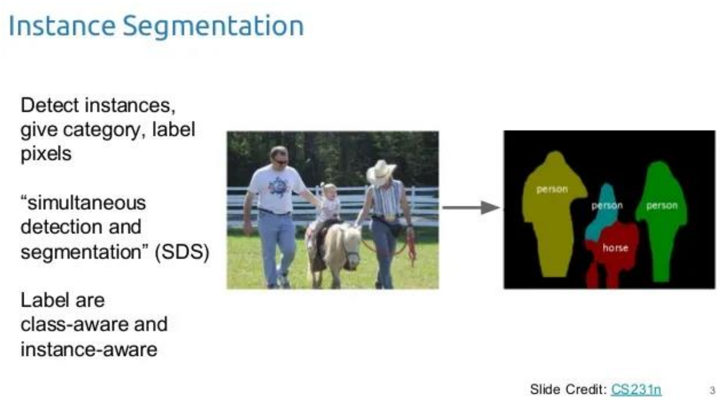
VGG-Net Architecture

- **Deep CNN developed by Visual Geometry Group (VGG) of Oxford university**
- **Task: Classification of color images belonging to 1000 classes in the ImageNet dataset**



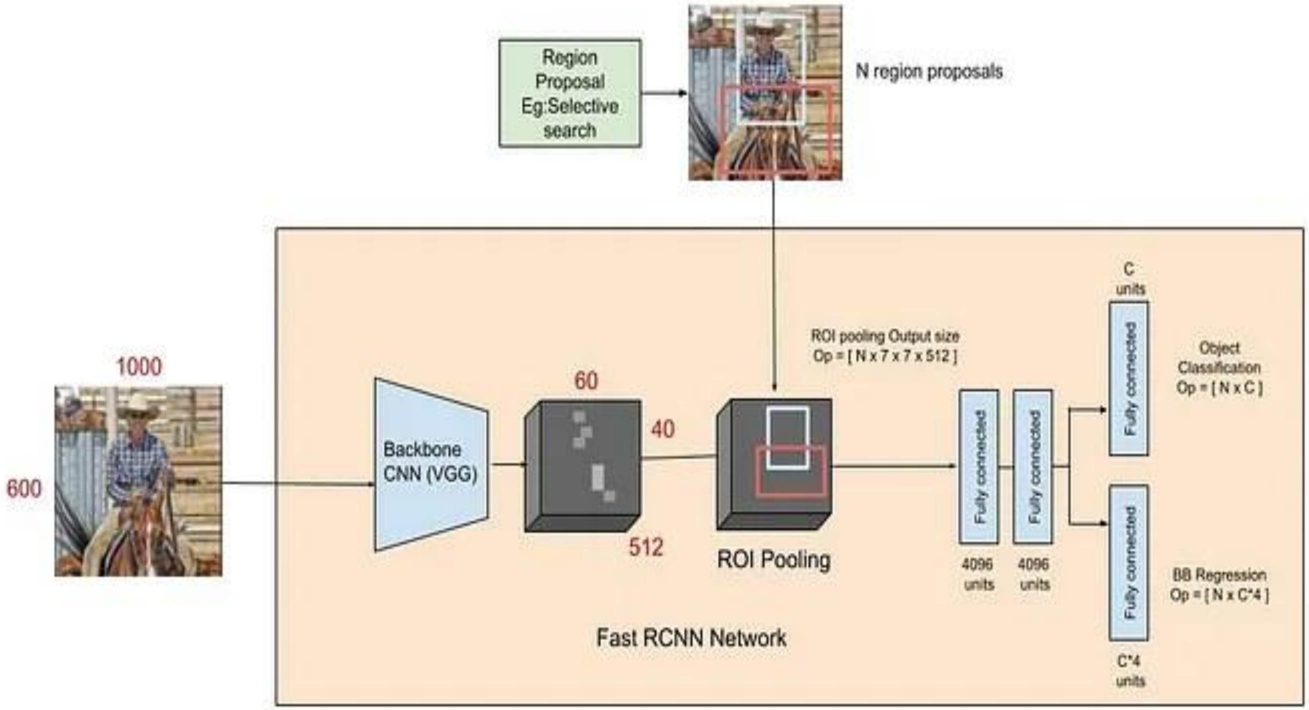
U-Net for Image Segmentation

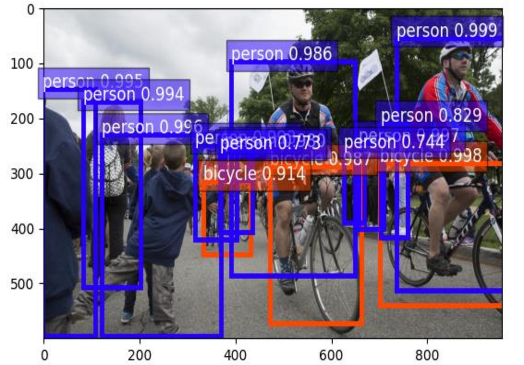




O.Ronneberger, P.Fischer, and T.Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation ", arXiv, 2015.

Faster Region-based CNN (Faster R-CNN) for Object Detection



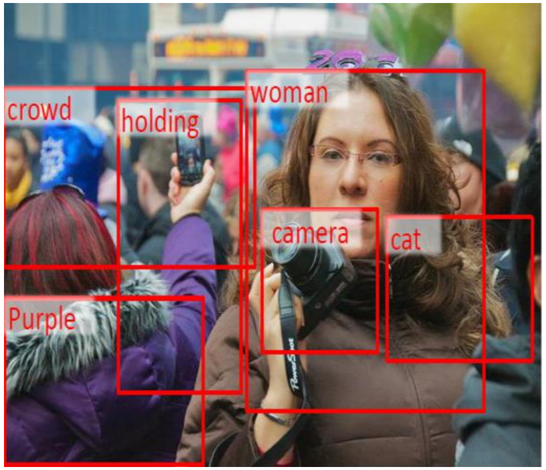


S.Ren, K.He, R.Girschick and J.Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, arXiv, 2016.

Image Captioning

A group of people at an outdoor market.

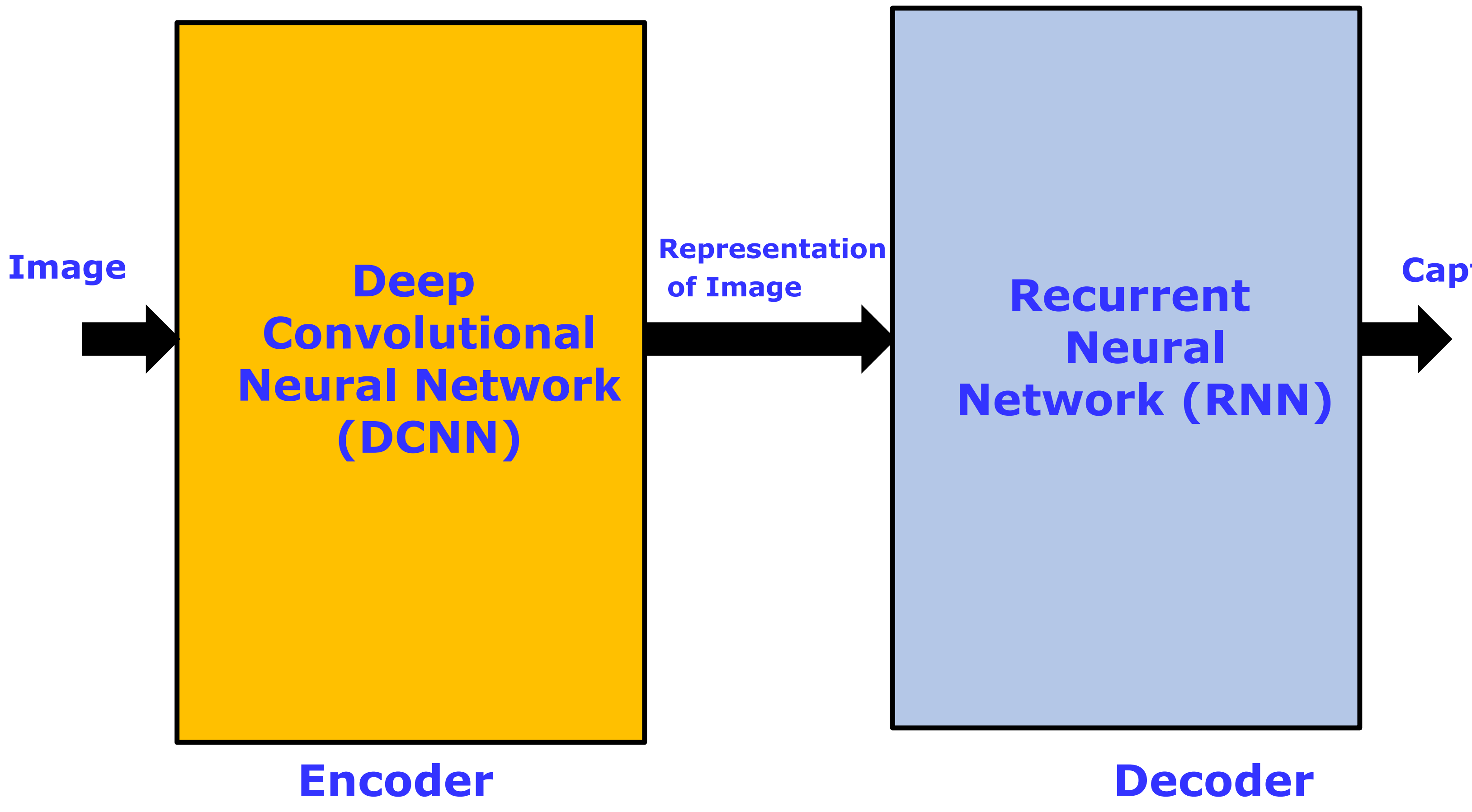
O. Vinyals, A. Toshev, S. Bengio and D.Erhan, "Show and tell: Lessons learned from the 2015 MSCOCO Image Captioning Challenge," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no.4, pp.652-663, April 2017.



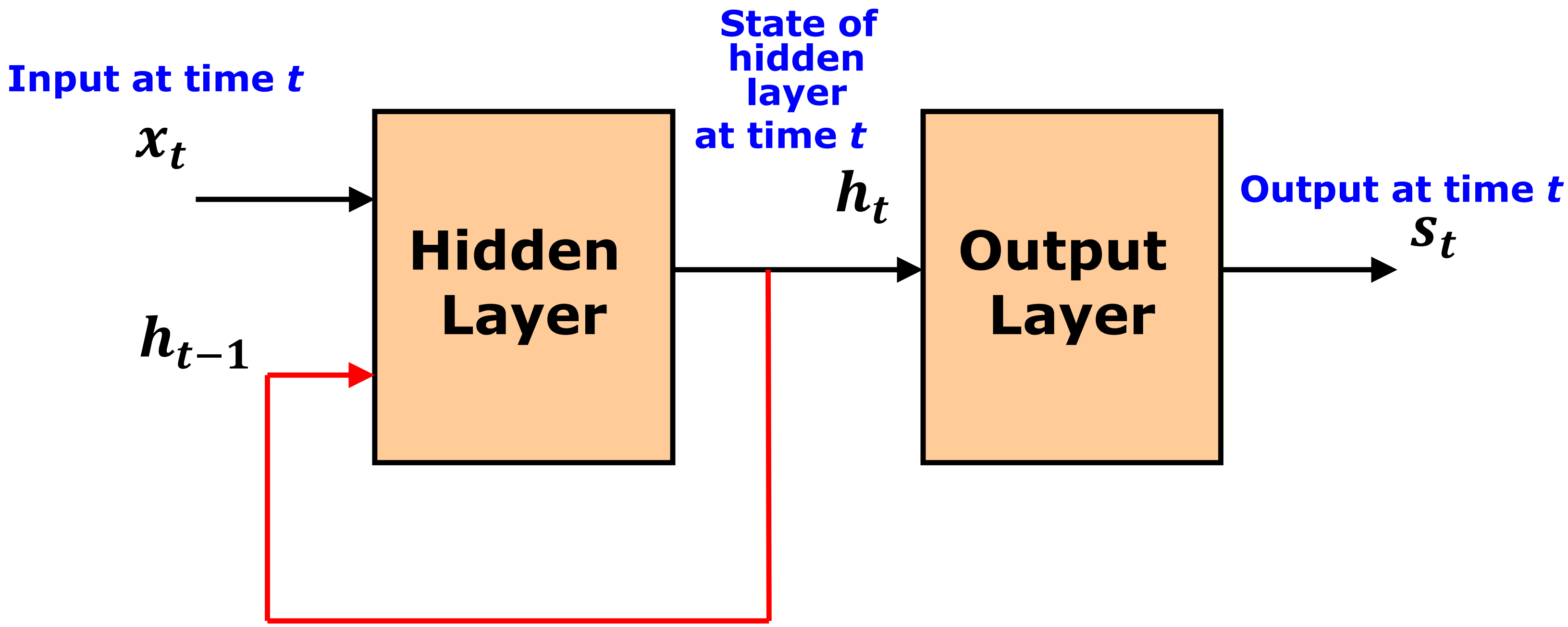
A woman holding a camera in crowd

Fang et al., "From captions to visual concepts and back, CVPR, 2015.

Encoder-Decoder Paradigm for Image Captioning



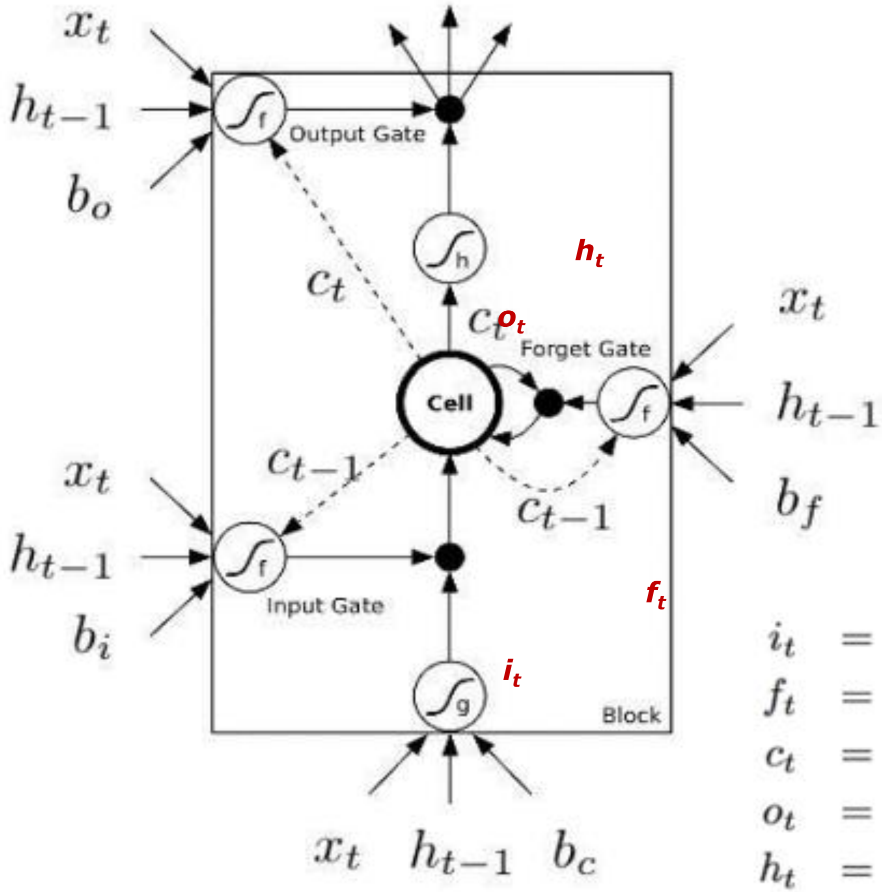
Recurrent Neural Network (RNN)



- **The hidden layer uses sigmoidal neurons**
- **The state of hidden layer (outputs of nodes in the hidden layer) at time t, h_t , is dependent on the input at time t and the state of the hidden layer at time t-1.**
- **The RNN that uses sigmoidal neurons in its hidden layer is shown to have the vanishing and exploding gradients problem, due to which the convergence during training is slow.**

Long Short-Term Memory (LSTM)

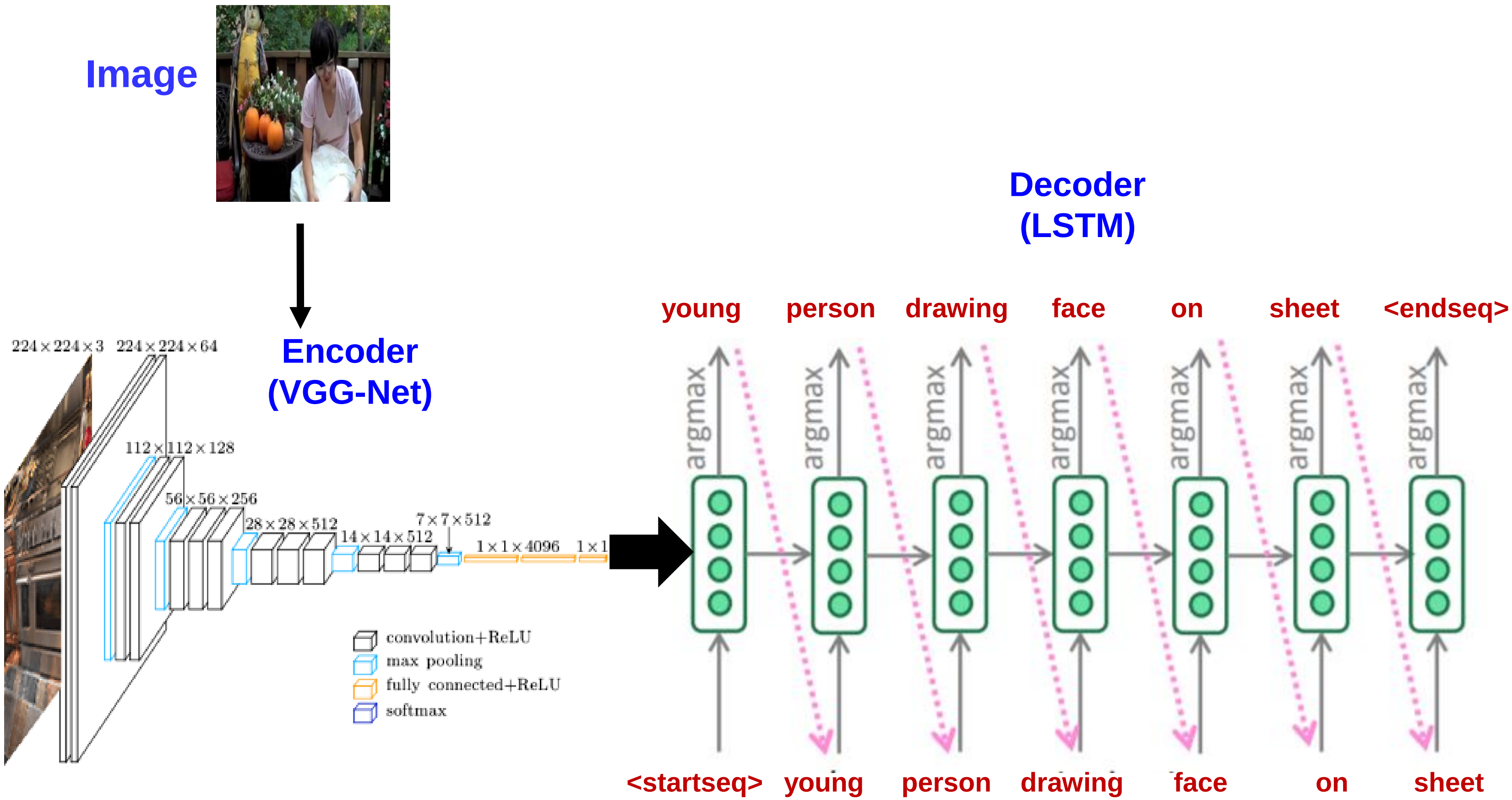
- **Structure of an LSTM Cell**



$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \\
c_t &= f_t c_{t-1} + i_t tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \\
h_t &= o_t tanh(c_t)
\end{aligned}
$$

- **The RNN that uses LSTM neurons in its hidden layer is shown to avoid the vanishing gradients problem, leading to faster convergence during training**

Encoder-Decoder Paradigm for Image Captioning



- The output of the pre-final layer in the CNN based encoder is used as the initial state of the hidden layer of LSTM based decoder

Embedding Methods

- **Image Embedding Methods**
  - **Output of pre-final layer of a deep CNN**
  - **Vector of Linearly Aggregated Descriptors (VLAD)**
  - **NetVLAD**

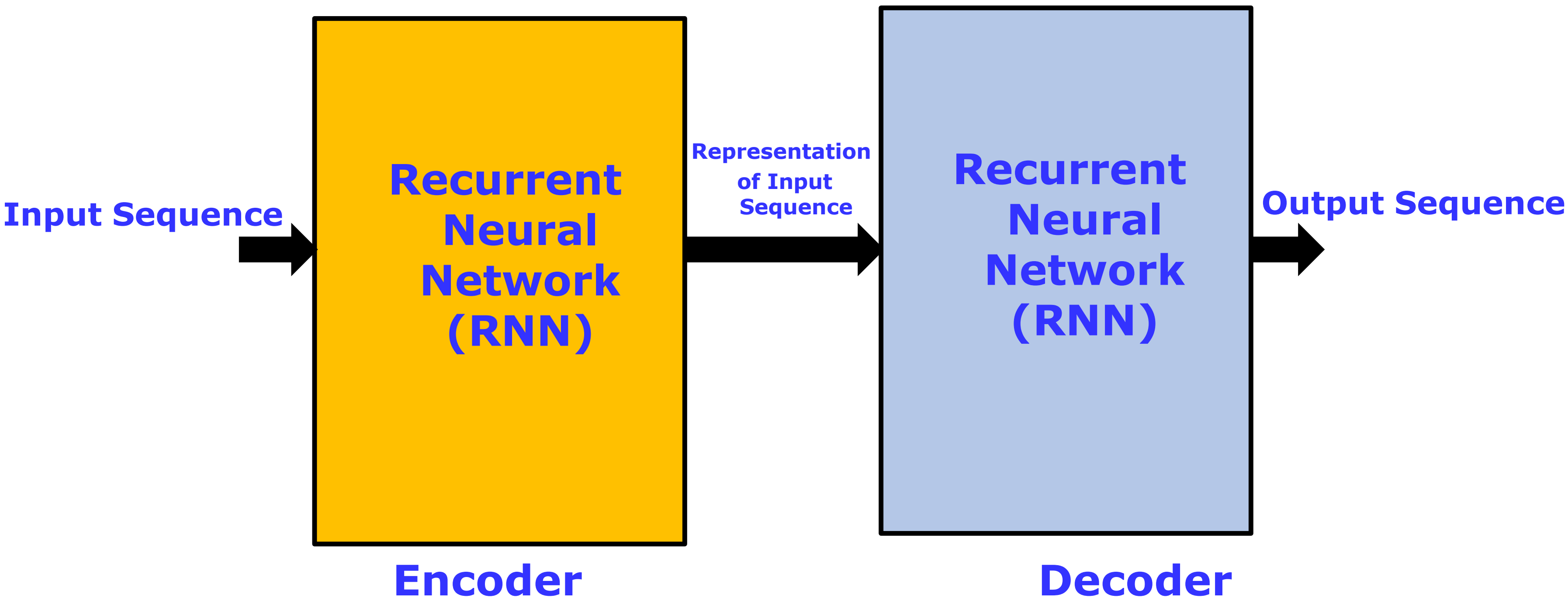
- **Video Embedding Method:**
  - **Sequential VLAD**

- **Word Embedding Methods**
  - **Word2Vec**
  - **GloVe**
  - **FastText**

Sequence-to-Sequence Mapping Tasks

- **Neural Machine Translation: Translation of a sentence in the source language to a sentence in the target language**
  - **Input: A sequence of words**
  - **Output: A sequence of words**
- **Video Captioning: Generation of a sentence as the caption for a video represented as a sequence of frames**
  - **Input: A sequence of feature vectors extracted from the frames of a video**
  - **Output: A sequence of words**
- **Each of the above tasks involves mapping an input sequence to an output sequence**

Encoder-Decoder Paradigm for Sequence-to-Sequence Mapping



Encoder-Decoder Paradigm for Sequence-to-Sequence Mapping

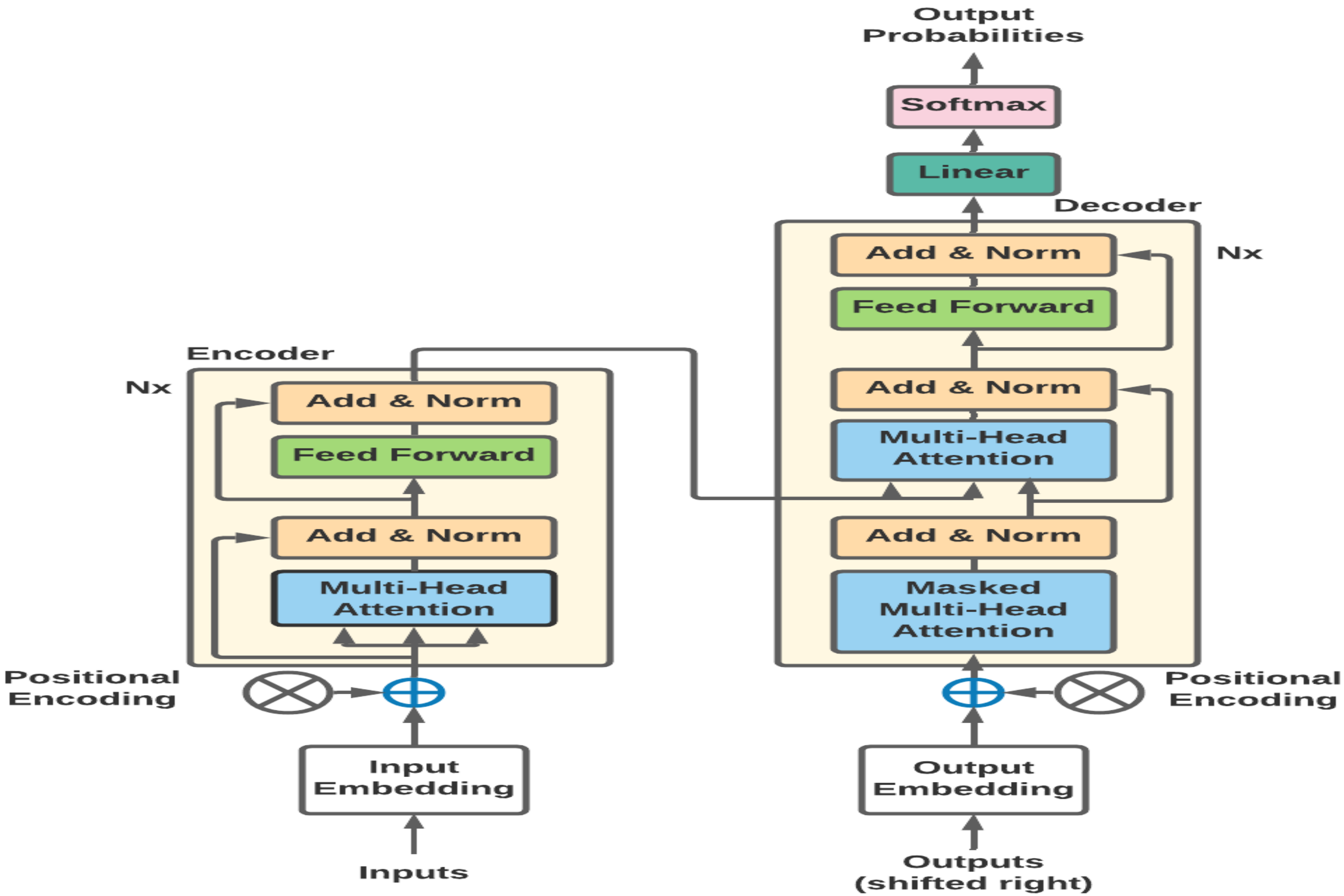
- **Sequence-to-Sequence Mapping using Encoder-Decoder Paradigm**
  - **Encoder: Generate a representation of the input sequence**
  - **Representation generated by Encoder is given as input to Decoder**
  - **Decoder: Generate the output sequence (A sequence of words)**
- **Relationship among the elements of a sequence:**
  - **Typically, an element in the input sequence is related to a few other elements in the input sequence**
  - **Typically, a word in the output sequence to be generated is related to a few elements in the input sequence**
- **LSTM based approach to Sequence-to-Sequence Mapping**
  - **Bidirectional LSTM based Encoder captures dependencies among elements in the input sequence**
  - **Bidirectional LSTM based Decoder captures dependencies among elements in the output sequence**
  - **Attention mechanism is introduced to capture dependencies of elements in the output sequence on elements in the input sequence**
- **Training the LSTM based Sequence-to-Sequence mapping systems is computationally intensive, and there is not much scope for parallelization of operations in the training process**

Attention based Models for Sequence-to-Sequence Mapping

- **Attention based models try to capture and use**
  - **Relations among elements in the input sequence (Self-Attention)**
  - **Relations among elements in the output sequence (Self-Attention)**
  - **Relations between elements in the input sequence and elements in the output sequence (Cross-Attention)**

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," NIPS, 2017.

Attention-based Model: Transformer



A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," NIPS, 2017.

Pre-training of Transformer

Encoder and/or decoder of transformer can be pre-trained using huge amount of unlabeled data, and then fine-tuned using small amount of labeled data for a downstream task.

- **Encoder pre-training for text data**
  - **Bidirectional Encoder Representation from Transformer (BERT)**

- **Decoder pre-training for text data**
  - **Generative Pre-trained Transformer (GPT)**

Bidirectional Encoder Representation from Transformer (BERT)

- **Pre-train the generic representation for several Natural Language Processing (NLP) tasks**

- **Pre-training Methods:**
  - **Masked Language Modelling (Mask LM)**
  - **Next Sentence Prediction (NSP)**

- **Fine-tuned for tasks such as**
  - **Sentence classification**
  - **Sentence relationship**
  - **Textual question answering**

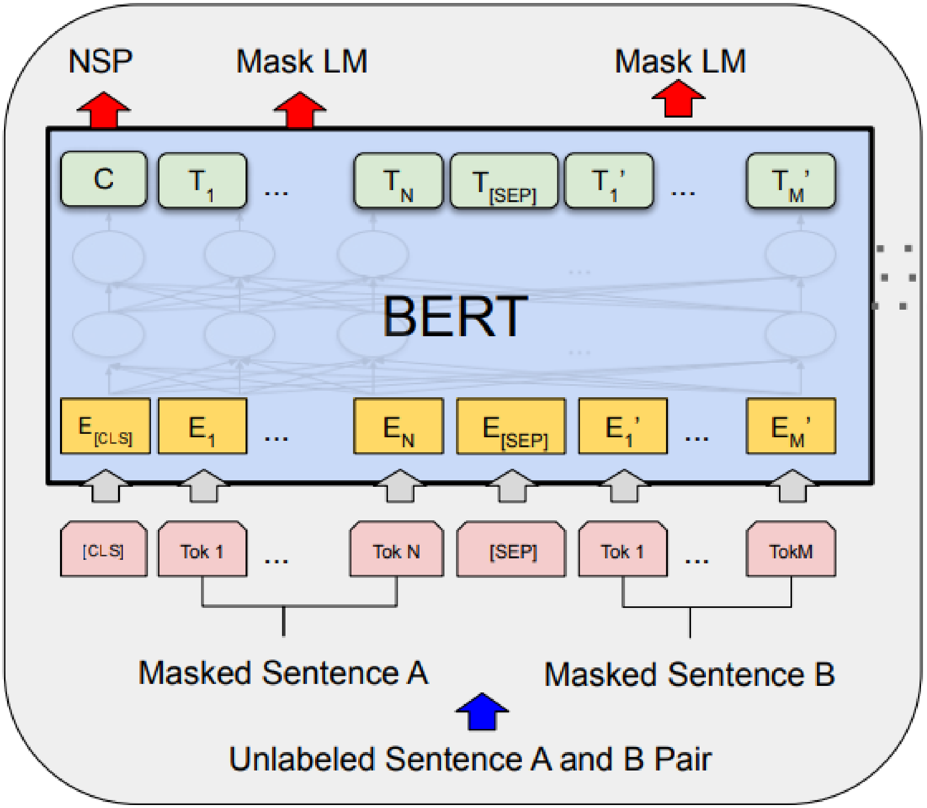


Image source : BERT(Devlin et al., 2019)

Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," NAACL, 2019.

Generative Pre-trained Transformer (GPT)

- Transformer decoder is pre-trained using unlabeled text data
- GPT can be fine-tuned for downstream tasks that involve text data
- Auto-regressive model: A word in a sentence is predicted using all the words preceding that word in the sentence
- Masked multi-head self-attention (MSA) in each layer of transformer decoder takes the sequence of words preceding a word in a sentence.
- The decoder is trained to predict the next word in the sentence.
- GPT-1, GPT-2 and GPT-3: Pre-trained models with different number of layers trained with different corpora for different pre-training tasks

A.Redford, K.Narasimhan, T.Salimans and I.Sutskever , "Improving Language Understanding by Generative Pre-training," 2018

A.Redford, J.Wu, R.Child, D.Luan, D.Amodei and I.Sutskever, "Language Models are Unsupervised Multitask Learners," 2019

T.Brown et al., "Language Models are Few-Shot Learners," arXiv:2005.14165v4, 22nd July, 2020

Visual Question Answering (VQA) for Images

Is there something to cut the vegetables with?

Yes

No

Who is wearing glasses?

Man

Woman

How many children are in the bed?

Two

One

Open Ended VQA

Question - What is the Zebra doing?
Traditional VQA - Eating, Grazing
Open Ended VQA - The Zebra is grazing in grasslands

Question - What is in the dog's mouth?
Traditional VQA - Toy, Purple toy
Open Ended VQA - The dog is playing with a toy in its mouth.

Image VQA Framework

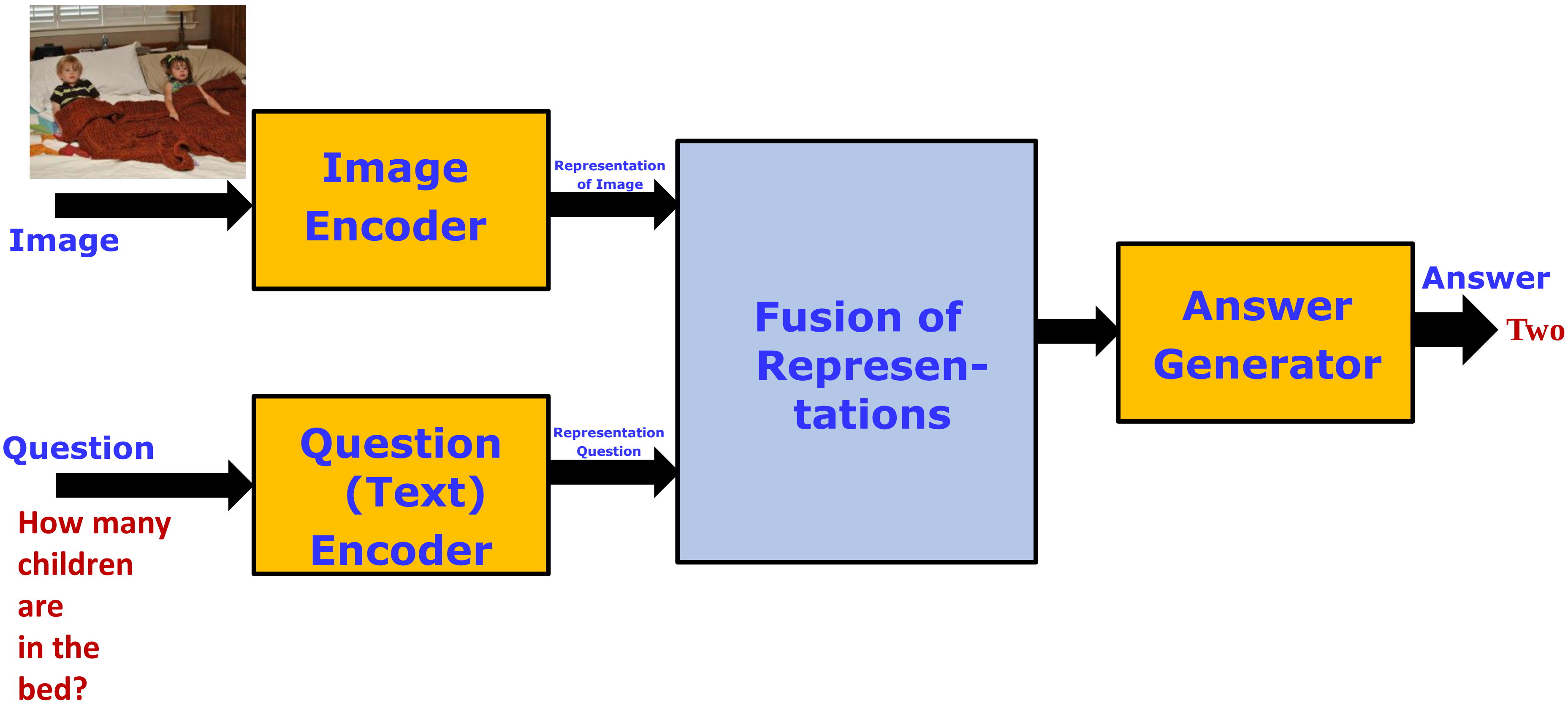


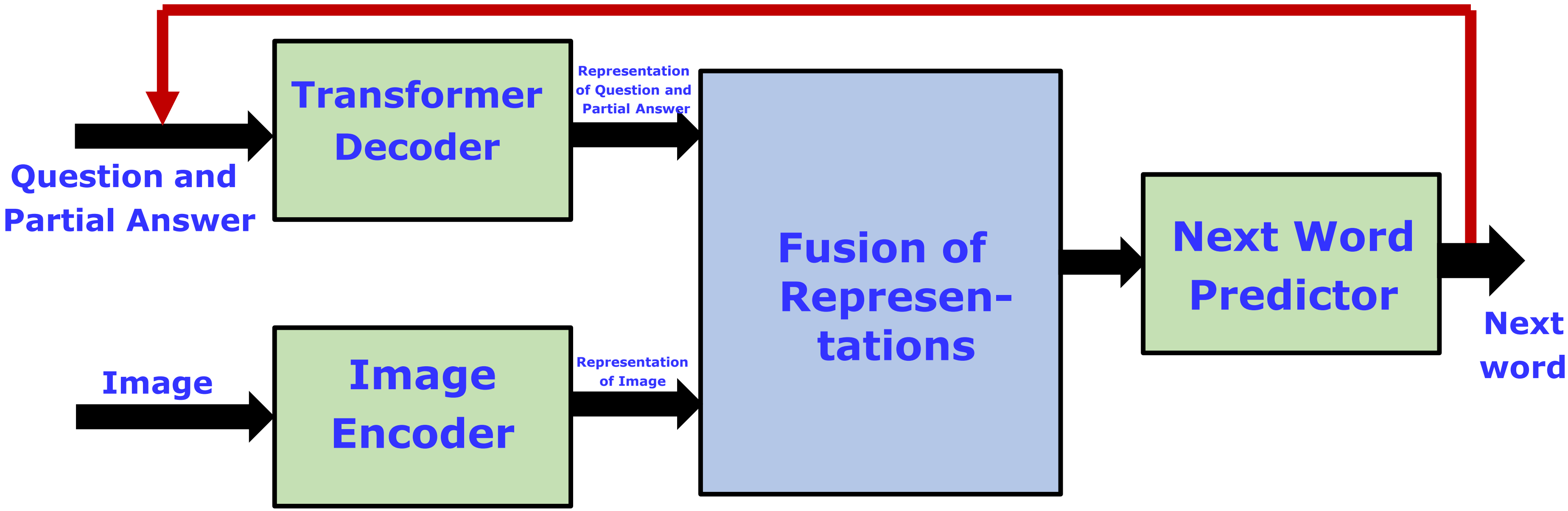
Image Encoder: CNN, ViT Encoder, Swin Tranformer

Question Encoder: LSTM, Transformer encoder, BERT fine-tuned with questions in VQA dataset

Fusion of Representations: Concatenation, Co-attention transformer

Answer Generator: Classifier, Text generator such as GPT fine-tuned with answers in VQA dataset

Open Ended VQA Framework



In open ended VQA, the answer is a sequence of words. The system generates one word of the answer at a time. The next word in the answer is predicted using the representations of image, question, and the partial answer corresponding to the sequence of words generated so far.

A.M.Bellini, N.Parde, M.Matteucci and M.J.Carman, "Towards Open-Ended VQA Models using Transformers," EMNLP, 2020.

Generative Models

- **Models capable of generation of data (Text, Image, Video, Music)**
- **Restricted Boltzmann machine (RBM)**
- **Variational autoencoder**
- **Generative pre-trained transformer (GPT)**
  - **Large Language Models (LLMs)**
- **Generative adversarial network (GAN)**
- **Diffusion models**
  - **Text-to-image**
  - **Text-to-video**
  - **Text-to-audio**
  - **Text-to-music**

LLMs: Evolution of GPT Models

1 — GPT-1

- Released: 2018
- 117 million parameters
- Trained using unsupervised learning
- Gave state of the art performance for LAMBADA.
- Gave Competitive performance for GLUE and SQuAD

2 — GPT-2

- Released: 2019
- Improved version of GPT-1.(1.5 billion Params)
- Trained on larger dataset.
- Additional improvements like including a modified training objective, and a more efficient sampling algorithm for generating text.

3 — GPT-3

- Released: 2020
- Improved version of GPT-2 (175 billion Params)
- Trained on larger dataset.
- Additional improvements like better training methodology (Gshard) and Few shot learning Capability.

4 — GPT-3.5

- Released:2022
- Improved version of GPT-3
- Three variants , 1.3 B, 6B and 175 B params
- The main element is to eliminate toxic outputs to certain extent using RLHF (reinforcement learning with human feedback).

5 — GPT-4

- Released: 2023
- Not sure if its a fine tuned version of previous model.
- 1 trillion Parameters
- It can take text and image as inputs and produces text outputs.
- RLHF is used in the training.

NLP Benchmarks:

LAMBADA: LAnguage Modeling Broadened to Account for Discourse Aspects

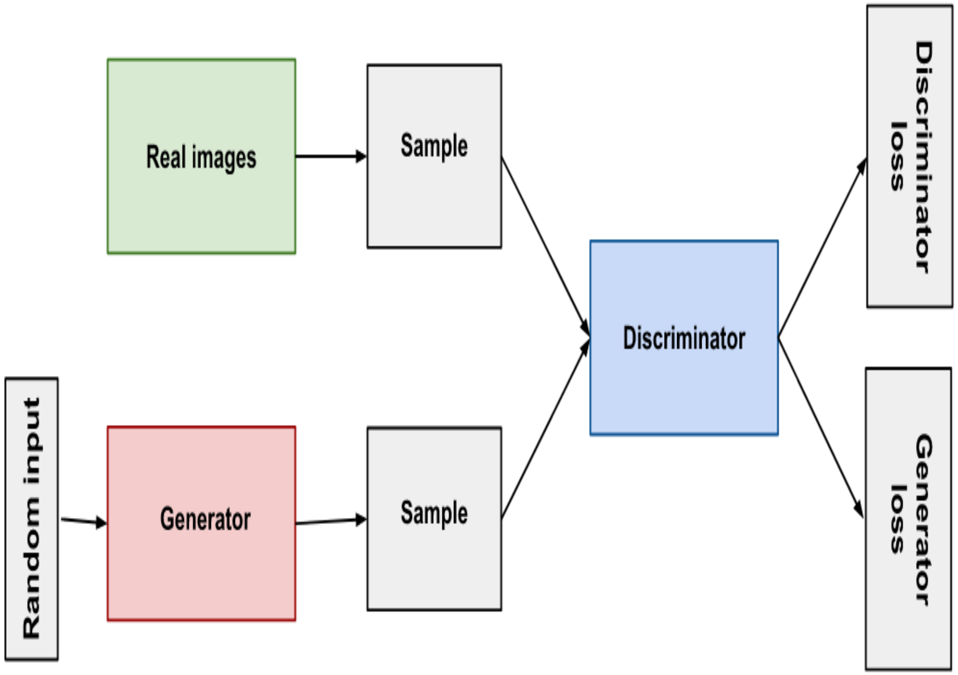
GLUE: General Language Understanding Evaluation

SQUaD: Stanford Question Answering Dataset

Image Generation using GAN

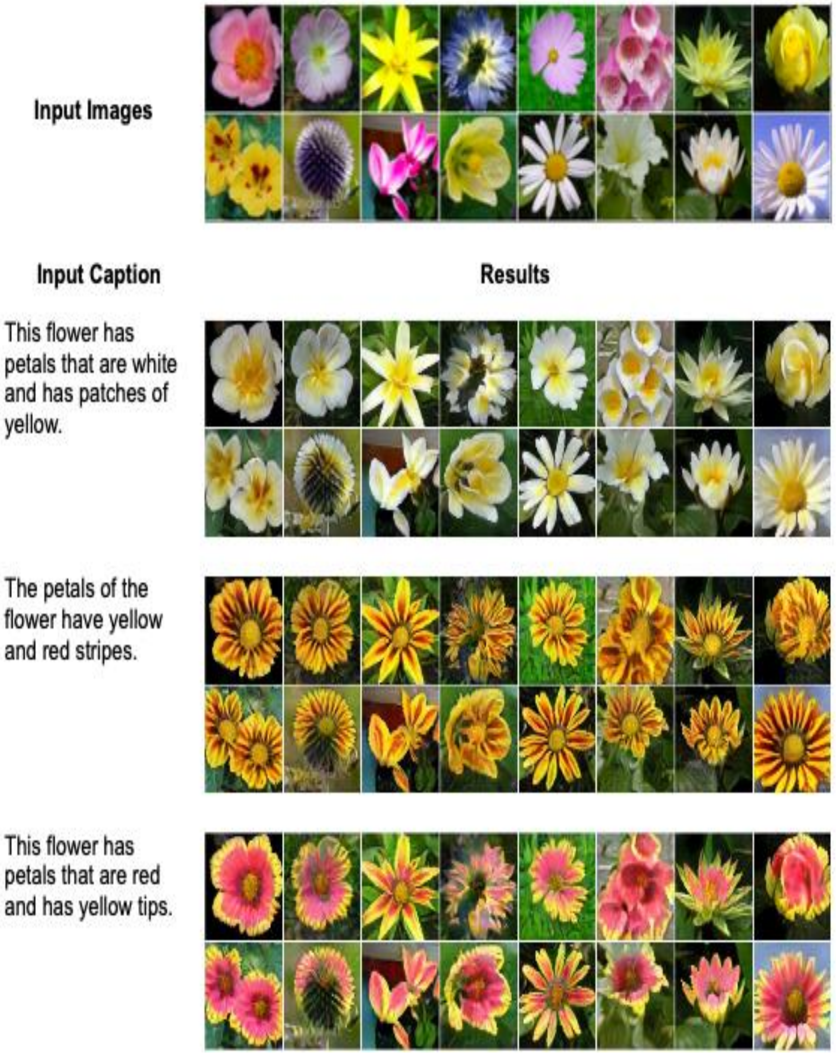
Generative Adversarial Network (GAN)

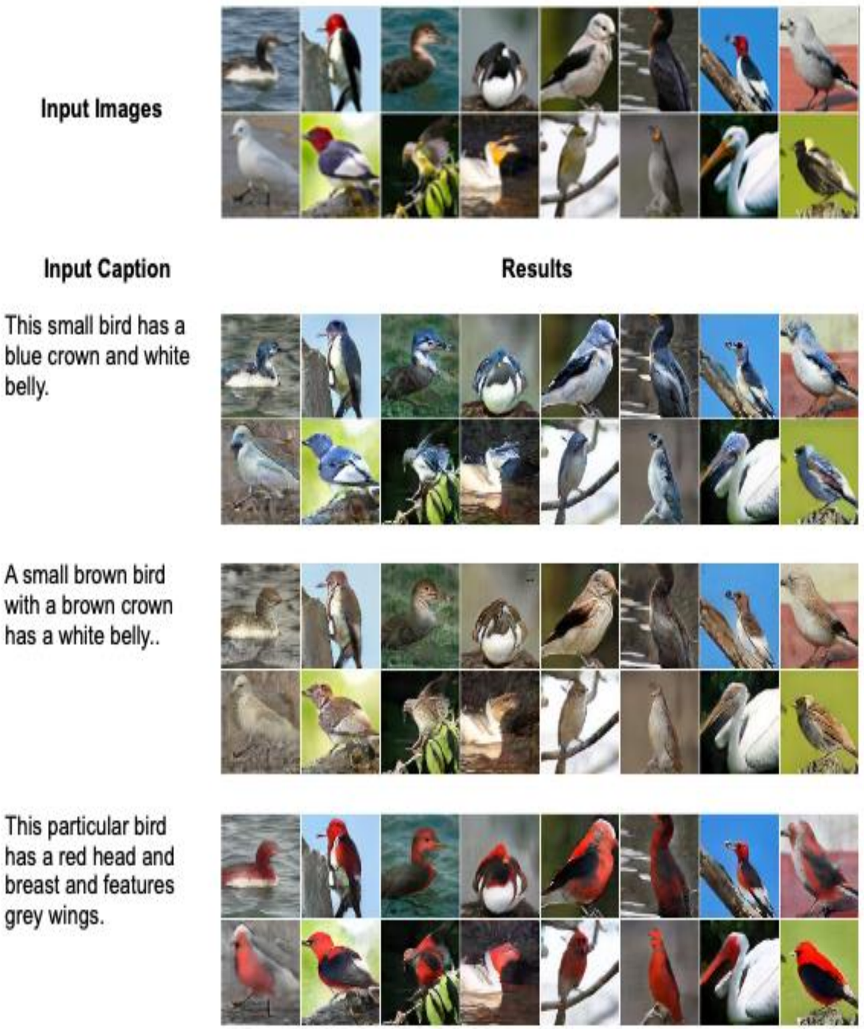
- **Generator and Discriminator are CNNs.**
- **Generator is a CNN with transposed convolution. It takes a random vector as input and generates an image as the output.**
- **Discriminator is a CNN based 2-class classifier that is trained to discriminate between the real images and the fake images generated by the Generator.**



Architecture of GAN

Image Manipulation using Text Adaptive GAN

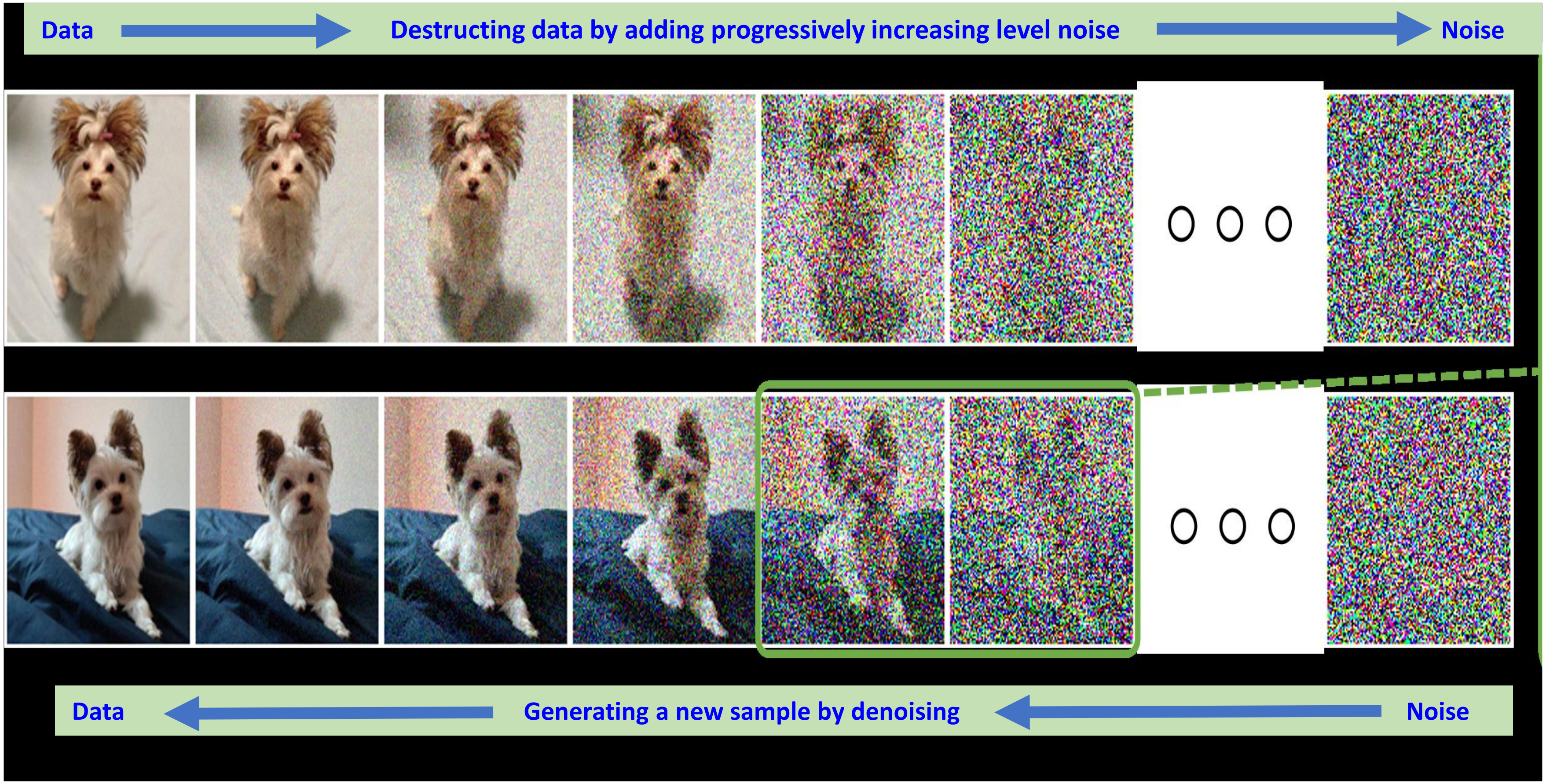




Variants of GAN Model

- Conditional GAN
- Cyclic GAN
- Cycle-Consistent GAN
- InstaGAN
- Progressive GAN
- Style GAN
- Self-Attention GAN
- BlockGAN
- GANFormer
- TextGAN

Denoising Diffusion Models for Image Generation



L.Yang et al., "Diffusion Models: A Comprehensive Survey of Methods and Applications," arXiv, 2023.

Coverage of Topics

1. **Introduction to deep learning**
2. **Feedforward neural networks:** Model of an artificial neuron, Activation functions: Sigmoidal function, Recti-linear unit (ReLU) function, Softmax function, Multi-layer feedforward neural network, Backpropagation method, Gradient descent method, Stochastic gradient descent method
3. **Optimization and regularization methods for deep feedforward neural networks (DFNNs):** Optimization methods: Generalized delta rule, AdaGrad, RMSProp, Adadelta, AdaM, Second order methods; Regularization methods: Dropout, Dropconnect; Batch normalization
4. **Autoencoders:** Autoassociative neural network, Stacked autoencoder, Greedy layer-wise training, Pre-training of a DFNN using a stacked autoencoder

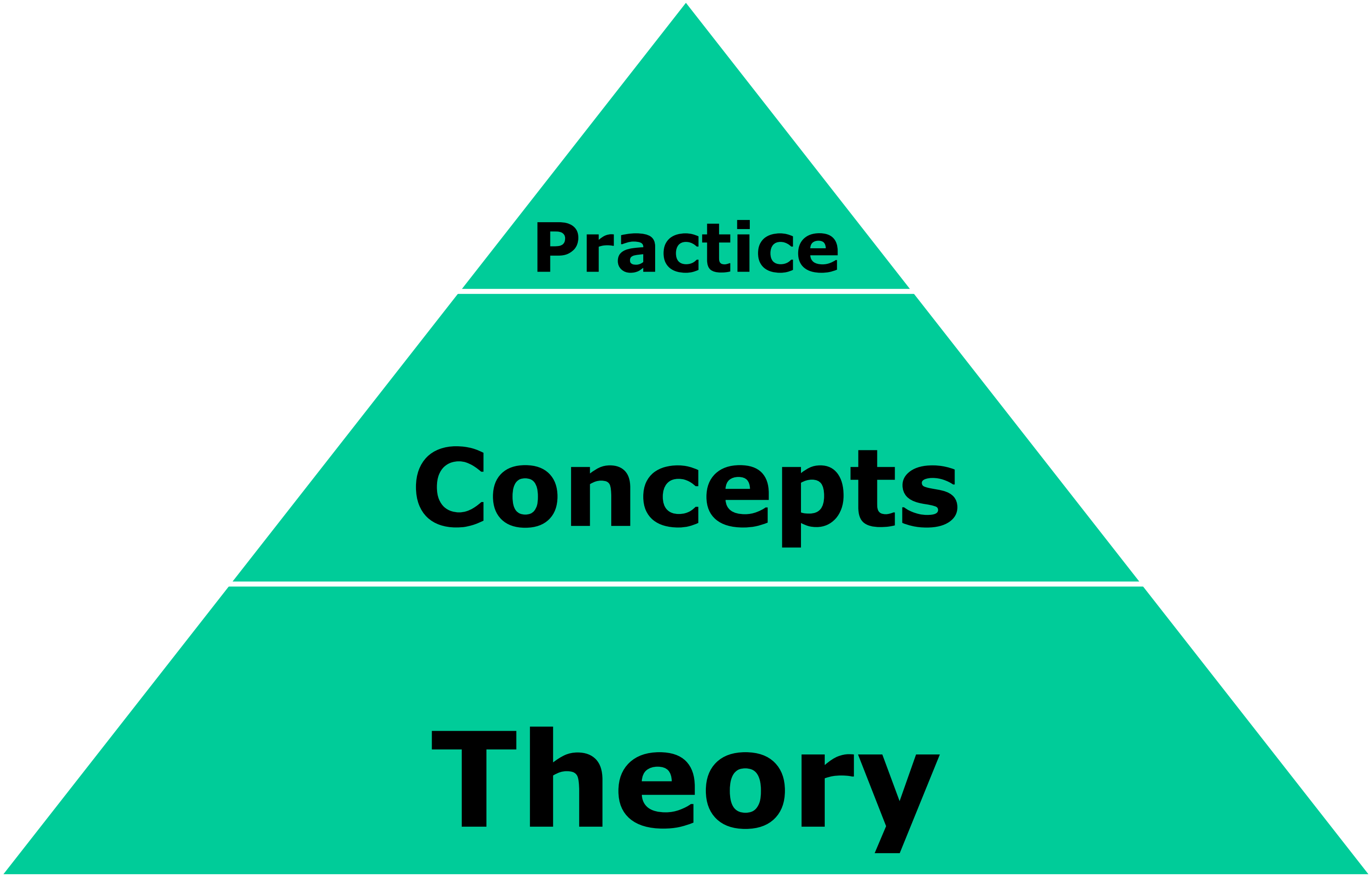
Coverage of Topics (Contd.)

5. **Convolutional neural networks (CNNs):** Basic CNN architecture, Deep CNNs for image classification: LeNet, VGGNet, GoogLeNet, ResNet; CNNs for image segmentation: U-Net and Fast RCNN; 1-d CNNs, 3-d CNNs
6. **Recurrent neural networks (RNNs):** Architecture of an RNN, Unfolding an RNN, Backpropagation through time, Vanishing and exploding gradient problems in RNNs, Long short term memory (LSTM) units, Gated recurrent units, Bidirectional RNNs
7. **Embedding methods:** Image and video embedding methods: VLAD, NetVLAD, Sequential VLAD; Word embedding methods: Word2Vec, GloVe, FastText

Coverage of Topics (Contd.)

8. **Transformer models:** Attention based models, Scale dot product attention, Multi-head attention (MHA), Self-attention MHA, Cross-attention MHA, Position encoding, Encoder module in a transformer, Decoder module in a transformer, Sequence to sequence mapping using transformer, Bidirectional encoder representations from transformers (BERT) model for text processing, Pre-training a BERT model, Fine-tuning, Generative pre-trained transformer (GPT), Introduction to large language models (LLMs)
9. **Generative Models:** Variational autoencoder, Generative adversarial networks (GANs), Introduction to diffusion models

Technology Pyramid



L.R.Rabiner and R.W.Schafer, Theory and Applications of Digital Speech Processing, Prentice Hall, 2011

Books and Evaluation Pattern

Text Books:

1. **C.M.Bishop and H.Bishop, Deep Learning: Foundations and Concepts, Springer, 2024**
2. **S.J.D.Prince, Understanding Deep Learning, MIT Press, 2023**
3. **I.Drori, The Science of Deep Learning, Cambridge University Press, 2022**

Reference Books:

1. **I.Goodfellow, Y.Bengio and A.Courville, Deep Learning, MIT Press, 2016**
2. **Charu C. Aggarwal, Neural Networks and Deep Learning, Springer, 2nd Ed., 2023**
3. **Nithin Buduma, Nikhil Buduma, Joe Papa, Fundamentals of Deep Learning, O'Reilly, 2nd Ed., 2022**

Evaluation Pattern (Tentative)

- **Programming Assignments: 30%**
- **Midsem Examination: 25% (90 Minutes, 15th March, 2024)**
- **Endsem Examination: 45% (180 Minutes, 9th May, 2024)**